



TREBALL FINAL DE GRAU



ESCOLA
POLITÈCNICA SUPERIOR
UNIVERSITAT DE LLEIDA
INSPIRING THE FUTURE

Estudiant: Marc Visa Pascual

Titulació: Grau en Enginyeria Informàtica

Títol de Treball Final de Grau: **Software per a la gestió del rendiment esportiu.**

Director/a: **Fernando Cores Prado**

Presentació

Mes: Juny

Any: 2020



SPORTS PERFORMANCE SOFTWARE MANAGEMENT

Treball Final de Grau



Autor:
Marc Visa Pascual

Codirectors:
Fernando Cores Prado,
Ángel Ric Díez

Grau en Enginyeria Informàtica
Curs 2019-2020

Índex de continguts

CAPÍTOL I.....	6
Introducció	6
1.1. Contextualització.....	6
1.1.1. Formulació del problema.....	6
1.1.2. Definició del Projecte	6
1.1.3. Marc del Projecte	7
1.2. Gestió del Treball	8
1.2.1. Objectius	8
1.2.2. Estructuració	9
1.3. Metodologia del Projecte.....	9
1.3.1. Model de procés de desenvolupament de software	9
1.4. Organització del Treball	11
1.4.1. Definició de tasques.....	11
1.4.2. Planificació Temporal.....	11
1.4.3. Pressupost	18
1.4.3.1. Taula recursos humans.....	18
1.5. Contingut de la memòria	20
CAPÍTOL II.....	21
Estudis previs al procés de desenvolupament del software	21
2.1. Estat de l'art	21
2.1.1. Articles	23
2.1.1.1. Data-Driven Visual Performance Analysis in Soccer.....	23
2.1.1.1.1. Variables Utilitzades.....	24
2.1.1.1.2. Flux de treball.....	26
2.1.1.2. Digital Analysis and Visualization of Swimming Motion	28
2.1.1.2.1. Captura de moviment	29
2.1.1.2.2. Parametrització	30
2.1.1.2.2. Visualització	30
2.2. Estudi dels fitxers de dades a gestionar.....	32
2.3. Selecció de la plataforma de desenvolupament i desplegament del projecte	33
2.3.1. Llenguatge R i Shiny	33

2.3.2. Node.js (<i>back-end</i>)	34
2.3.3. Angular (<i>front-end</i>)	35
CAPÍTOL III.....	37
Anàlisis i disseny del projecte	37
3.1. <i>Blast-off</i> del Projecte	37
3.1.1. Directrius del Projecte – raons i motivacions pel Projecte.....	38
3.1.1.1. El propòsit del Projecte	38
3.1.1.2. Implicats en el producte	39
3.1.1.3. Usuaris del producte	40
3.1.2. Restriccions del projecte i del producte.....	40
3.1.2.1. Restriccions als requisits.....	40
3.1.2.2. Acords de noms i definicions.....	41
3.1.2.3. Fets rellevants i suposicions	43
3.1.3. Requisits Funcionals – la funcionalitat del producte.....	43
3.1.3.1. L’abast del treball	43
3.1.3.2. L’abast del producte	43
3.1.4. Requisits No Funcionals – les qualitats del producte	44
3.1.4.1. Requisits d’usabilitat	44
3.1.4.2. Requisits de rendiment	44
3.1.4.3. Requisits legals	45
3.2. Business UseCases, ProductUseCases, <i>User-Stories</i>	45
3.2.1. The Brown Cow Model.....	46
3.2.1.1. BUCs i PUCs.....	47
3.2.1.1.1. Selecció variables de rendiment.....	47
3.2.1.1.2. Filtratge de variables de rendiment	49
3.2.1.1.3. Anàlisi i visualització de les variables de rendiment.	51
3.2.1.1.4. ...S’exposen les dades de les variables escollides referents als esdeveniments esportius capturats en una gràfica de línies.	53
3.2.2. User-Stories	54
3.2.3. Prototips	59
3.2.4. Refinament dels requeriments	61
CAPÍTOL IV.....	63
Fase de desenvolupament	63
4. Presa de decisions per al desenvolupament del software	63

4.1.	<i>Framework</i> i llenguatges	63
4.2.	Bases de dades	63
4.3.	Control de versions	64
4.4.	Esquema de desplegament.....	64
4.5.	Patrons de disseny utilitzats.....	65
4.6.	Diagrames de disseny.....	66
4.6.1.	Diagrama de classes.....	66
4.6.2.	Diagrama de flux de treball	67
4.7.	Implementació.....	68
4.7.1.	Organització del Projecte software.....	68
4.7.1.1.	Fitxers del Projecte (directori TFG)	69
4.7.1.2.	Fitxers de l'aplicació (directori app).....	69
4.7.1.3.	Fitxers de desplegament.....	70
4.7.2.	API's externes utilitzades en el Projecte software.....	71
4.7.2.1.	Python Data AnalysisLibrary – <i>pandas</i>	71
4.7.2.2.	Python <i>math</i> Idem	72
4.7.2.3.	Python <i>json</i> Idem	72
4.7.3.	Consideracions durant la implementació	72
CAPÍTOL V	82
Fase de validació	82
5.1.	Tests del Projecte	82
5.1.1.	Ad-hoc Testing	82
5.1.2.	<i>Checklists</i> per la validació i qualitat del Software.....	83
CAPÍTOL VI	89
Conclusions	89
6.1.	Conclusions finals	89
6.2.	Resultats	91
6.3.	Treball futur	91
Bibliografia	93

Índex de figures

Figura 1. Variant del model en cascada amb retroalimentació i prototipat.	10
Figura 2. Diagrama de Gantt de l'anàlisi de requisits.	12
Figura 3. Diagrama de Gantt fase de desenvolupament.	14
Figura 4. Diagrama de Gantt fase de validació	16
Figura 5. Diagrama de Gantt memòria del Treball	17
Figura 6. Exemple mapa de calor.....	26
Figura 7. Flux de treball article anàlisi futbol.	27
Figura 8. Exemple captura digital de moviment.....	29
Figura 9. Exemple tracking format silueta.	31
Figura 10. Brown Cow Model.	46
Figura 11. Prototip configuració anàlisi.	59
Figura 12. Prototip carregar fitxers de dades.	60
Figura 14. Prototip selector de variables de rendiment.	61
Figura 15. Diagrama de classes UML	66
Figura 16. Diagrama de flux de l'aplicació web.	68
Figura 17. Formulari paràmetres de configuració.....	73
Figura 18. Formulari, per fitxers esdeveniments, de noms clau i noms de les variables de rendiment per representar en el gràfic de dades.	74
Figura 19. Formulari noms clau per fitxers de dispositius	74
Figura 20. Llista de fitxers carregats durant la sessió	75
Figura 21. Formulari per carregar fitxer de dades	75
Figura 22. Taula de dades + selector de variables	76
Figura 23. Algorisme de filtratge de dades segons configuració de usuari	77
Figura 24. Algorisme conversió de float a enter	78
Figura 25. Interpolació de fitxers d'esdeveniments	78
Figura 26. Interpolació de fitxers de dispositius.....	79
Figura 27. Downsampling.....	79
Figura 28. Algorisme representació dades en la taula html.....	80
Figura 29. Algorisme per crear el gràfic de dades.....	81
Figura 30. Creació d'un gràfic gràcies a Chart.js	81

Índex de taules

Taula 1. Total de costos de recursos humans.....	19
Taula 2. Checksit validació qualitat de la base de dades	84
Taula 3. Checklist validació funcionalitats de la web.....	88

CAPÍTOL I

Introducció

1.1. Contextualització

1.1.1. Formulació del problema

Desenvolupament d'una plataforma informàtica que permeti gestionar i visualitzar variables de rendiment derivades de dades posicionals extretes des de sistemes de *tracking* en esports col·lectius. Per això s'haurà de realitzar la integració de dades de diferents dispositius (GPS, UWB, acceleròmetre, giroscopi, etc.) en un programari o suport web per analitzar i estudiar el comportament col·lectiu i patrons de comportament entre els diferents integrants d'un equip de futbol, parella de pàdel o grup dansa, entre d'altres.

1.1.2. Definició del Projecte

Aquest projecte consistirà en un procés de desenvolupament d'una aplicació/plataforma web, la qual llegirà paràmetres d'entrada (dades capturades en format text (seqüències de *strings*) de fitxers *csv*), realitzant un procés d'integració entre elles per permetre un anàlisi posterior.

Les dades s'organitzaran a partir d'una, o més d'una, freqüència de mostreig determinada, és a dir, un nombre de mostres per unitat de temps preses d'una senyal contínua, en aquest cas els fotogrames o *frames* per segon, i en un període de temps mitjançant un formulari. Aquestes seran estructurades en una taula per visualitzar-les, ordenar-les segons un criteri o condició en funció de l'anàlisi de dades que es vulgui dur a terme.

Es podran seleccionar els tipus de dades (variables de rendiment) amb un control de formulari en forma de caselles de verificació per la seva posterior representació en la taula.

1.1.3. Marc del Projecte

El Projecte que es desenvoluparà al llarg d'aquest Treball Final de Grau estarà ambientat i focalitzat en la base tecnològica de l'entrenament d'alt rendiment de l'esportista.

L'esport d'alt nivell ha anat de la mà amb les noves tecnologies de diferents àmbits. Al llarg dels últims anys s'ha donat molta importància en processos de control de l'entrenament i la competició. En innumbrables estudis o treballs d'investigació s'han utilitzat variables "externes" com poden ser temps, distàncies, velocitats, potències, etc. I altres variables "internes" com ara la freqüència cardíaca, pressió arterial, consum d'oxigen a altes o baixes intensitats, etc.

En els entrenaments o competicions d'alt rendiment esportiu, el fet de millorar els resultats, pot ser degut a petits canvis de posició, moviment, etc., que només es poden detectar i corregir mitjançant les tecnologies de la informació (TIC).

La introducció de les tecnologies IT, a través de la captura de dades per mitjà de sensors estratègicament col·locats en els/les esportistes que volen perfeccionar les seves marques, tècniques, coneixements, etc., ha estat un dels grans èxits en els centres d'alt rendiment (CAR).

Aquest Projecte s'emmarca en aquest sector o àmbit de la tecnologia, informàtica i enginyeria. Pretén donar més llum a aquests sistemes de millora esportiva gràcies a les TIC.

En relació a les mètriques que es prenen dins de l'anàlisi del rendiment esportiu són les referències específiques a varies qualitats físiques per a cada esport. Això inclou la força, la velocitat, la flexibilitat, etc., indispensables per l'obtenció

d'un rendiment òptim. Aquestes mètriques [35] són determinants per unes proves atlètiques. Alguns exemples com:

- Dades antropomètriques: alçada, pes, composició corporal.
- Força muscular (tren superior i tren inferior).
- Velocitat (acceleració, esprint, resistència).
- Agilitat (control de moviment i equilibri, coordinació de mètriques anteriors).

Amb aquestes mètriques, bàsicament:

1. Es dona un punt de partida de la condició física de l'atleta.
2. S'ajuda a saber quin és el nivell de l'atleta de la seva competitivitat i a poder-lo millorar o augmentar.
3. S'identifica el tipus d'entrenament individual de l'atleta amb les dades del seu rendiment.
4. Es preven lesions.

1.2. Gestió del Treball

En aquest apartat, es presentaran els objectius que s'han marcat per al projecte a desenvolupar en aquest Treball. A més a més, s'explicarà quina ha sigut l'estructura per seguir el desenvolupament del Treball, com s'ha dividit i el perquè s'ha volgut efectuar d'aquesta manera.

1.2.1. Objectius

Ara s'esmentaran els objectius per aconseguir una bona utilitat per al projecte així com aconseguir realitzar un desenvolupament efectiu d'aquest.

Dins el projecte desenvolupat es poden definir 2 objectius principals:

- Desenvolupar una aplicació web útil per l'anàlisi del rendiment d'un equip esportiu o professional en l'esport.

- **Backend** (la capa d'accés a dades d'un software o qualsevol dispositiu no accessible directament als usuaris)
- **Frontend** (totes les tecnologies de disseny i desenvolupament web que corren en el navegador i que s'encarreguen de la interactivitat amb els usuaris)
- Gestionar, integrar i visualitzar variables de rendiment derivades de dades posicionals extretes de sistemes de tracking (GPS, UWB, acceleròmetre, giroscopi, etc.) en esports col·lectius.

1.2.2. Estructuració

S'ha optat per estructurar el Treball d'acord amb les diferents fases que té un procés de desenvolupament de software a part de seccions teòriques, de recerca o, bàsicament, pròpies d'un Treball d'alt nivell. Així, a mesura que s'avançava en el Projecte, si sorgien requisits nous, noves formes d'implementar diferents funcionalitats del producte, alternatives a l'hora de validar el sistema i/o sub-sistemes implementats, etc. Simplement s'anava afegint el nou contingut en el capítol o capítols del Treball. Per cada fase del procés de desenvolupament del software s'introdueixen conceptes particulars de tècniques, mètodes o temes generals per poder entrar en context i poder abordar cada fase amb èxit.

1.3. Metodologia del Projecte

1.3.1. Model de procés de desenvolupament de software

Per a poder estudiar de forma genèrica el procés de desenvolupament de software del Projecte del Treball, la opció més adient ha estat el model en cascada amb retroalimentació i prototipat. Aquest model, que es mostra en la

figura 1, ordena rigorosament les fases del desenvolupament del software, de tal manera que l'inici de cada fase és en finalitzar la fase anterior.

Gràcies a aquest model, el prototipat ha aportat una avaluació i una millor descripció dels requeriments, és a dir, refinar-los, capturar nous requeriments i provar opcions de disseny. Tanmateix, pot ser temptador desenvolupar a partir del prototip, però podria arrossegar decisions de disseny inapropiades, per tant no s'ha de tenir com una prioritat alta a seguir durant les fases de disseny i implementació, ja que pot crear desconfiança en el client del producte.

Encara que la retroalimentació també estigui en les fases inicials el qual dóna una reducció en la rigidesa del model clàssic, continua sent molt localitzada. A més a més, aquest model facilita la correcció de l'especificació, reduint la propagació d'errors i examina la viabilitat i la utilitat del software.

En definitiva, s'ha decidit per aquest model ja que resulta més útil per qualsevol tipus de software i perquè a l'avançar de fase en fase es pot tornar enrere si sorgeix cap assumpte, permetent així una gestió de qualsevol problema de manera efectiva.

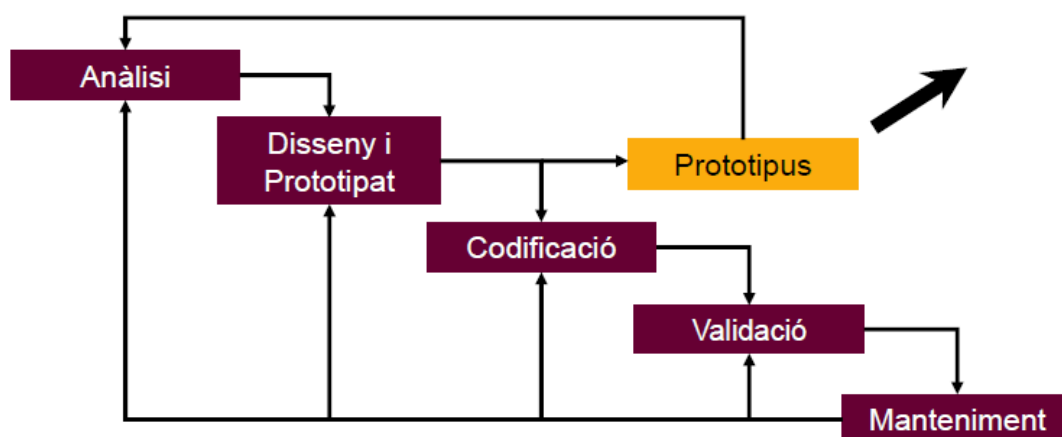


Figura 1. Variant del model en cascada amb retroalimentació i prototipat.

1.4. Organització del Treball

Es definiran les tasques per al projecte com també una planificació inicial per executar aquestes i, malgrat complicacions, problemes o contratemps, una segona planificació per les tasques inicials que no s'han pogut finalitzar a temps igual que tasques noves que han hagut de sorgir durant el desenvolupament del projecte.

1.4.1. Definició de tasques

S'ha decidit que el projecte es divideixi en les següents tasques:

- Documentació
- Anàlisi de requeriments
- Disseny i arquitectura de l'aplicació
- Implementació del disseny
- Proves de test per verificar la funcionalitat de l'aplicació
- Conclusions

A continuació es detalla cadascuna d'aquestes tasques i se'ls assigna una planificació temporal.

1.4.2. Planificació Temporal

Tasca d'Anàlisi

Una vegada s'enfocarà el Treball per abordar-lo, es començarà amb la fase d'anàlisi. Aquesta secció serà adient per estudiar el Projecte que s'hauria de desenvolupar. Es descobriran totes les necessitats del producte, tan funcionalitats com restriccions, es definirà el vocabulari específic pel projecte, el propòsit i els abasts tan del projecte com del producte.

La planificació per aquesta fase tant destacada en el procés de desenvolupament de software es mostra en la **Figura 2**. Aquesta fase haurà de durar els mesos d'octubre i de novembre de 2019. La sub-tasca del Blast-off començarà el 5 d'octubre i acabarà el 9 del mateix mes. La sub-tasca de la realització dels BUCs, PUCs, UserStories i prototips s'iniciarà el 24 d'octubre i finalitzarà el 19 de novembre.

Com a segona planificació per aquesta tasca, s'ha afegirà una nova sub-tasca per a refinar els requeriments donat els canvis i adaptacions que s'han trobat al llarg del projecte. Aquesta sub-tasca es realitzarà el 24 de febrer.

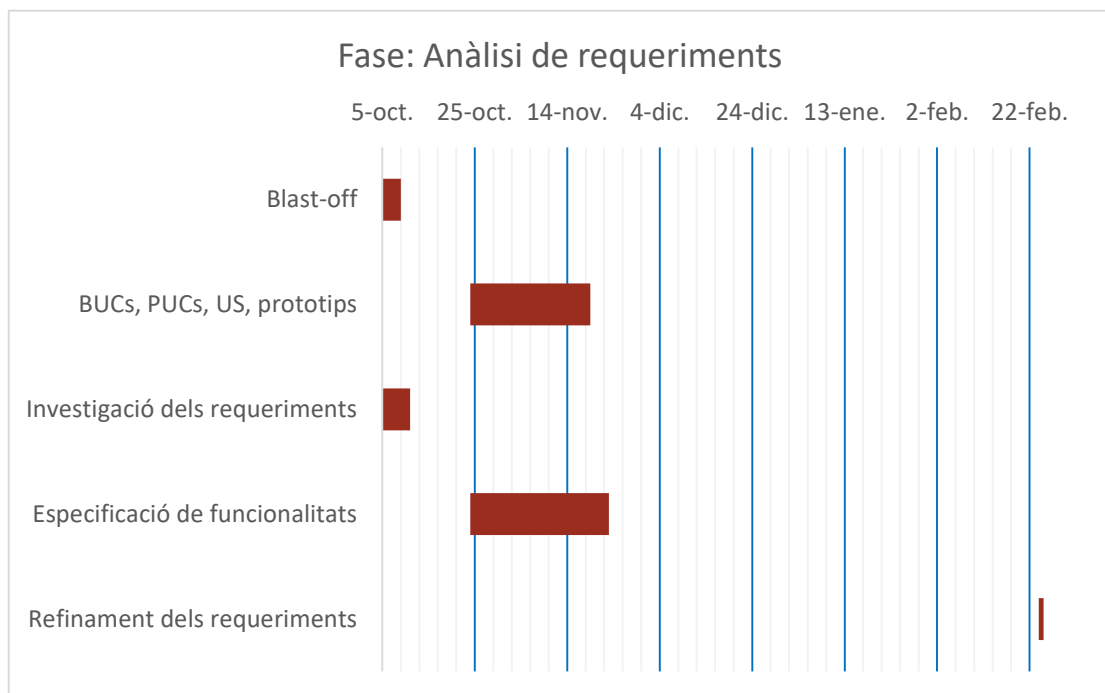


Figura 2. Diagrama de Gantt de l'anàlisi de requisits.

Tasques de Disseny, definició arquitectura i Implementació

En aquesta tasca del Treball es transmetrà totes les decisions de cara a focalitzar el Projecte. Tan decidir quines tecnologies s'utilitzaran de cara a la implementació de l'aplicació, com l'arquitectura de programari, quina serà la base de dades, la plataforma en núvol on es desplegarà l'aplicació i així oferir-la al públic, entre altres menesters. A més a més, també es necessitarà esbossar diagrames per plasmar les idees generals de l'estructura primordial de l'aplicació software.

Tenint el disseny precís, entenedor i intel·ligible, posteriorment s'exposaran els fitxers clau que componen el Projecte software i la funció de cadascun dins el funcionament del *framework* utilitzat perquè al final s'executi la web, com també els fitxers que es requereixen pel desplegament de la web, els de virtualització de l'aplicació, entre altres.

La planificació per aquesta fase tant destacada en el procés de desenvolupament de software es mostra en la **Figura 3** Aleshores, les fases de disseny i implementació durarà els mesos desembre de 2019, gener i febrer de 2020. Del 20 de desembre al 24 de desembre es realitzarà el Disseny de la Base de Dades. Del 27 de desembre al 3 de gener s'estructurarà de manera bàsica l'aplicació; es començarà desenvolupant el *backend* del 27 de desembre fins al 30, les taules i camps inicials de la BDD i la lògica inicial de les vistes principals; el 2 de gener s'iniciarà la navegació bàsica per la web i la presentació inicial de la lògica del *backend* de cada vista com a sub-tasca de *frontend* inicial.

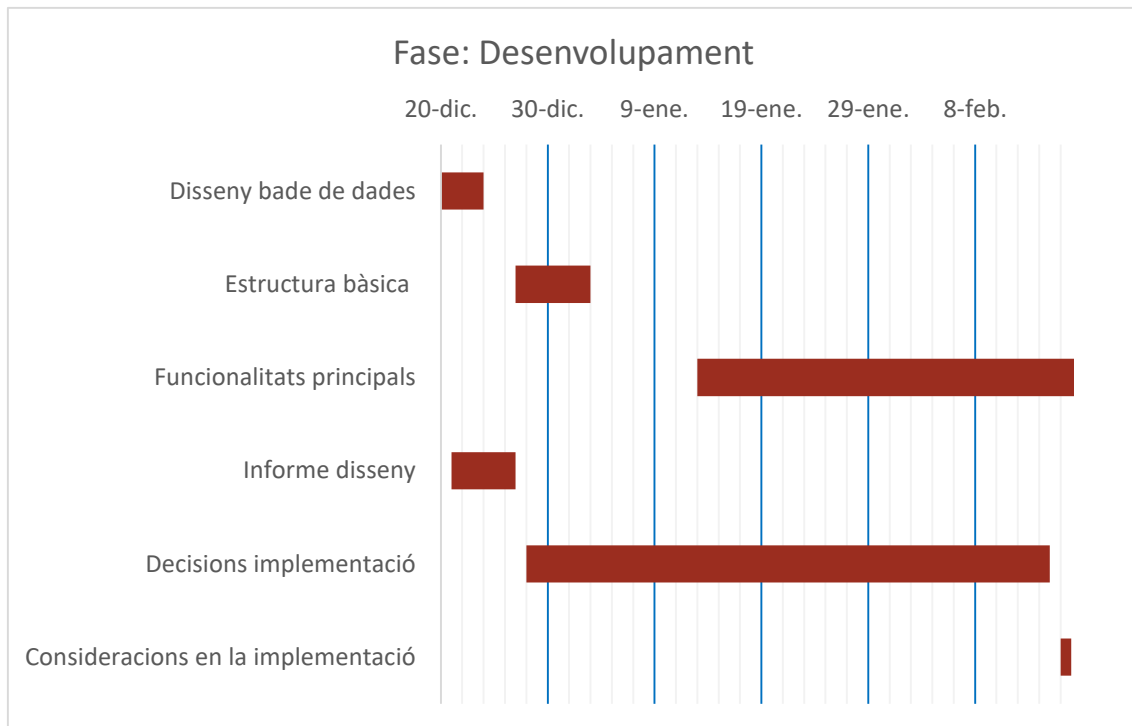


Figura 3. Diagrama de Gantt fase de desenvolupament.

Del 13 de gener al 28 de febrer s'implementaran les funcionalitats principals de l'aplicació. El 13 de gener es començarà amb la lògica de negoci de cada vista de l'aplicació, es processaran els formats de cada tipus de csv, s'interpolaran les dades del csv obtingut de la BD, es realitzarà *downsample* de les dades segons la freqüència de la configuració (de la taula de dades) introduïda en la BD, es filtraran les dades segons la configuració (de la taula de dades) introduïda en la BD. Tots aquests apartats hauran de finalitzar el 25 de febrer

Com a segona planificació per aquesta sub-tasca del *backend*, s'afegirà un nou apartat per a millores i tractament d'errors que durarà des del 26 de febrer al 28 de febrer.

El *frontend* principal (la lògica per tractar les dades processades del *backend* i representar-les) durarà del 29 de gener al 28 de febrer; en aquest període es construirà el selector amb les opcions corresponents segons les dades processades en el *backend*, es construirà la taula amb les dades processades i

filtrades del *backend* segons les opcions triades en el selector, s'exportarà les dades representades en la taula en format csv.

Com a segona planificació per aquesta sub-tasca del *frontend*, s'afegirà un nou apartat per a millores per a la presentació de les dades que durarà del 29 de febrer a l'1 de març.

Tasca de Validació (Testing)

En aquest sector del treball s'explicarà detalladament el tipus de test emprat i les eines utilitzades referents en aquest durant tota la construcció del Projecte de programari.

La planificació per aquesta fase tant destacada en el procés de desenvolupament de software es mostra en la **figura 4**. Aquesta fase s'executarà durant els mesos de desembre de 2019, gener i febrer, com a conseqüència de la durada de la tasca d'Implementació.

Del 28 de desembre fins al 4 de gener es comprovaran les funcionalitats bàsiques de l'aplicació tals com: la creació d'usuaris, que la navegació de la web sigui fluida, si s'emmagatzemen correctament les dades introduïdes a través de la UI (formularis) a la BD.

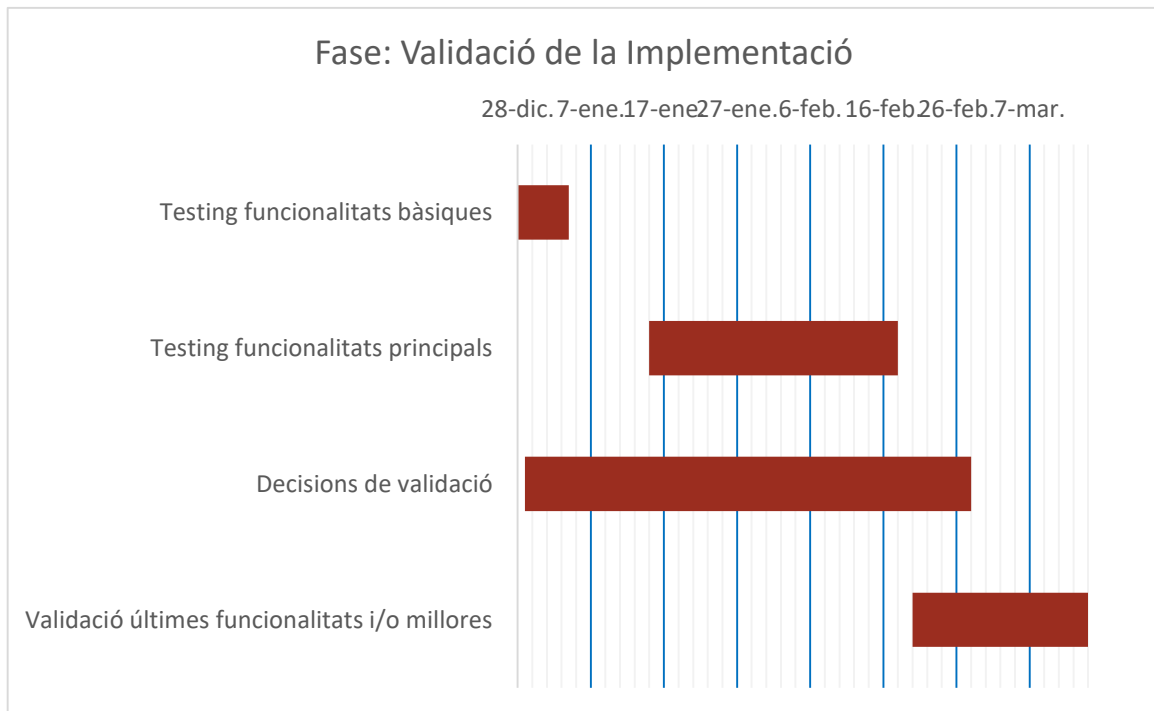


Figura 4. Diagrama de Gantt fase de validació

La comprovació de les funcionalitats principals durarà del 15 de gener al 18 de febrer (tan la lògica del *backend* com del *frontend*).

Com a segona planificació per aquesta tasca, s'afegirà una nova sub-tasca per a validar les últimes funcionalitats i millores esperades. Aquesta sub-tasca durarà des del 20 de febrer al 15 de març.

Documentació

Des de que es s'iniciarà el Treball, totes les recerques, possibles estructures i formats de document (taula de continguts, formats de títols de capítols, sub-capítols, etc.) de la memòria es plasmaran en format digital. Igualment, en cada fase de desenvolupament definida, s'informarà/documentarà de tota la realització de la fase, tan la realització de les seves tasques com les possibles alternatives si s'havien escaigut complicacions durant el seu transcurs.

La planificació per aquesta fase tant destacada en el procés de desenvolupament de software es mostra en la **Figura5**. Aleshores, aquesta última fase, la qual es realitzarà en paral·lel amb les altres, durarà de l'octubre del 2019 al març del 2020. Documentar la investigació dels requeriments (blast-off) del 5 a l'11 d'octubre, la especificació de funcionalitats (BUCs, PUCs, UserStories i prototips) del 24 d'octubre al 23 de novembre. La recerca de fonts d'interès (els estudis dels 2 articles) durarà del 10 al 23 d'octubre.

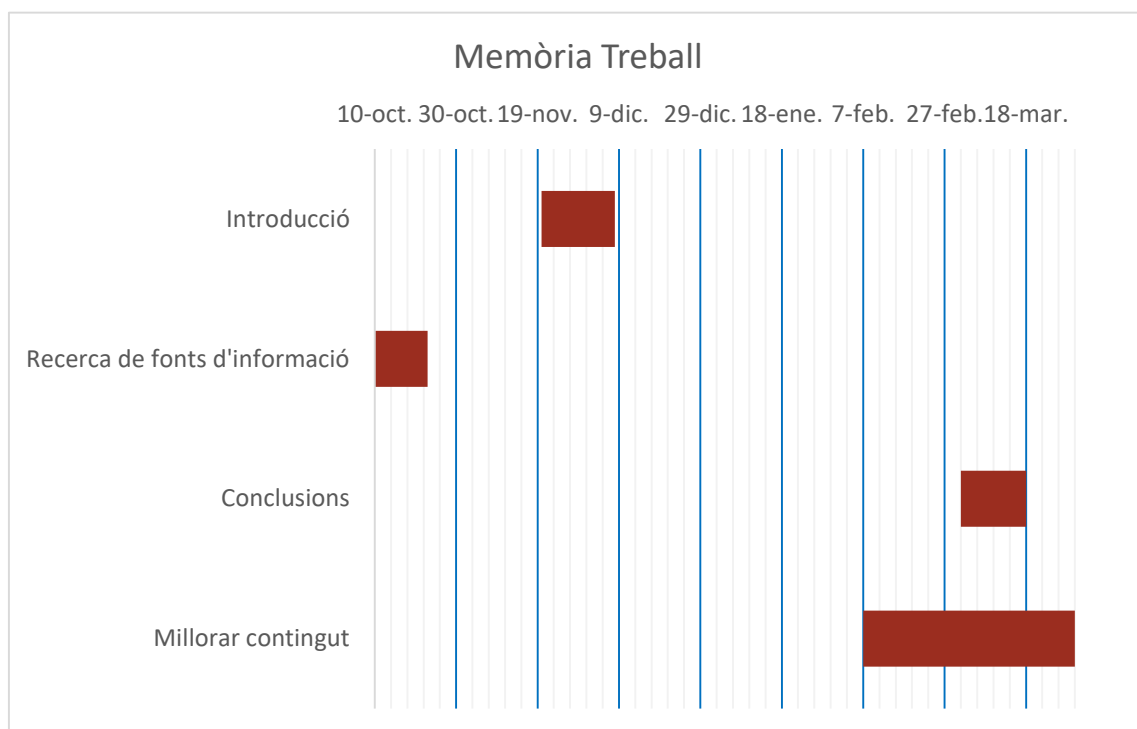


Figura 5. Diagrama de Gantt memòria del Treball

La Introducció s'iniciarà el 20 de novembre i acabarà el 8 de desembre, en la qual es realitzarà l'Estat de l'art, el Marc del projecte, la Definició del projecte, la Formulació del problema i la Metodologia del projecte.

L'informe del disseny de l'aplicació el qual indica la Bases de dades, l'Esquema de desplegament i el Flux de treball de l'aplicació durarà del 21 de desembre al 27 de desembre.

Del 28 de desembre al 15 de febrer es realitzarà l'Estudi tecnologies pel desenvolupament del projecte, Decidir els patrons de disseny a utilitzar i l'Estudi de API's utilitzades com a Decisions d'implementació.

Les decisions de la Validació (tipus de testing i les Quality Assurance Software Testing Checklists) s'iniciaran el 29 de desembre fins al 28 de febrer.

Donats els contratemps i adaptacions que ha sofert el projecte, es destinarà el 16 de febrer per una sub-tasca per a documentar les consideracions de la implementació i un període de temps del 7 de febrer al 30 de març, com a marge per a pròxims problemes, per a millorar el contingut de la memòria.

Les conclusions es començaran el 2 de març i acabaran el 18 de març.

Respecte les adaptacions mencionades anteriorment, el 12 d'abril es començarà una sub-tasca, donat que el client ha afegit un nou requisit a l'aplicació, una nova funcionalitat. Es destinarà del 14 d'abril al 22 d'abril a l'anàlisi d'aquest nou requeriment, la seva implementació en l'aplicació i, en conseqüència, la validació de la nova funcionalitat.

1.4.3. Pressupost

L'autor del treball assumeix que el cost de l'equip de treball (ordinador) més tot el software utilitzat li suposa un cost 0 per al pressupost del projecte. Tal i com es mostra en la **Taula 1**, els costos directes del projecte en relació amb els recursos humans.

1.4.3.1. Taula recursos humans

Rol	Sou	Hores/dia	Temps total (dies)	Cost
Analista	18 € / h	7	65	8190
Desenvolupador	16 € / h	8.5	45	6120

Quality Manager	20 € / h	8	25	4000
-----------------	----------	---	----	------

Taula 1. Total de costos de recursos humans

Al temps total que ha durat cada fase amb cada professional se li ha restat els dies de festa, és a dir, un total de 20 dies.

Així, el cost total per aquest projecte és de 18310 €.

Recursos Hardware

Les eines hardware que s'han utilitzat durant el Treball són:

- ASUS VivoBook Pro N752VX
 - Processador Intel(R) Core(TM) i7-6700HQ
 - Memòria RAM 16,0 GB
 - Disc dur 2TB (5400 rpm SATA3)
 - Controladora gràfica NVIDIA(R) GeForce(R) GTX 950M 2GB/4GB DDR3 VRAM

Recursos Software

Les eines software que s'han utilitzat durant el Treball són:

- Windows 10 Home: Sistema Operatiu de Microsoft.
- Microsoft Office Word, processador de textos comercialitzat per la companyia Microsoft.
- AdobeAcrobat Reader DC, per convertir el document Word a PDF.
- Microsoft Office Excel, programa de full de càlcul per crear els diagrames de Gantt.
- JetBrainsPyCharm Professional Edition, per desenvolupar el Projecte software.

1.5. Contingut de la memòria

El capítol II tractarà sobre la investigació que s'ha realitzat durant tot el Treball tan en la recerca d'informació per contextualitzar el Projecte i definir els requeriments, en estudiar les possibilitats i maneres d'implementar l'aplicació i les seves funcionalitats.

En el capítol III s'exposaran tots els documents d'anàlisi de requeriments que s'han fixat per aquest Projecte com també una explicació breu de les tècniques emprades en aquesta fase del procés de desenvolupament del software.

En el capítol IV es desglossarà tots els apartats de la fase de desenvolupament del procés de software. Es comentaran les decisions preses tan en el disseny i modelatge de l'aplicació com en les pràctiques de la implementació (*framework/s* junt amb els llenguatges de programació utilitzats i les diferents *APIs* emprades). A més a més, s'explicarà també totes les consideracions que s'han pres durant la implementació de l'aplicació i com s'han gestionat en la codificació.

En el capítol V s'explicarà la finalitat de la fase de validació, el tipus de *testing* que s'ha fet servir per comprovar el bon funcionament de l'aplicació, juntament amb la metodologia emprada per la validació i revisió del software.

En el capítol VI es mostraran unes conclusions generals per part de l'autor del Treball acompanyades d'uns resultats finals obtinguts del Projecte i un possible futur per al projecte, noves possibles funcionalitats i/o modificacions. A més a més, s'exposaran totes les fonts d'informació que s'han consultat durant tot el Treball com a bibliografia, web-grafia.

CAPÍTOL II

Estudis previs al procés de desenvolupament del software

2.1. Estat de l'art

El món de l'esport cada dia s'està tornant més tecnològic degut a la combinació del talent atlètic amb els anàlisis de nivell avançat i la intel·ligència artificial, amb la fi d'obtenir els millors resultats possibles en el camp esportiu.

“ La tecnologia juga un paper molt important en el desenvolupament de l'esport i, d'aquesta manera, contribueix en la millora del rendiment en totes les seves facetes. Per això, les aplicacions tecnològiques permeten realitzar un entrenament més efectiu, estimulacions musculars, tractament i seguiment d'atletes, precisió en els resultats, incrementar el rendiment i prevenir lesions ”(Busch, 1998).

Les tecnologies avantguardistes [25] s'utilitzen per optimitzar el rendiment en diversos esports com el ciclisme, la natació, el golf, l'esquí, el surf, el futbol, el tennis, etc. Aquestes tecnologies de l'esport es classifiquen en sis categories.

1. Les **auto-tecnologies** són les que alteren la constitució física i psicològica de l'esportista, com ara els fàrmacs (o mètodes) de millora de rendiment (dopatge). També s'inclouen processos quirúrgics, pròtesis bioniques, psicologia esportiva, enginyeria genètica.
2. Les **tecnologies panoràmiques** estan relacionades en l'entorn esportiu, el qual inclou la manera en què l'espectador/a percep els esdeveniments esportius. L'augment de complexes esportius multifuncionals amb pantalles i cúpules retràctils, càmeres elevades, superfícies

(prefabricades) de pistes (seguiment continu del rendiment) i gespa artificial. El sistema de posició global (GPS) pot ser utilitzat juntament amb sensors mecànics (acceleròmetres, giroscopis, etc.) per avaluar i monitoritzar l'activitat física. El GPS també pot ser combinat amb sensors biomèdics, meteorològics (termòmetre, baròmetre, etc.); com una evolució més en l'anàlisi del rendiment.

3. Les **tecnologies d'equipament** inclouen els equips que els/les atletes utilitzen dins la seva pròpia disciplina. Per exemple: cascs en el futbol amb dispositius d'alerta i ràdios, vestits "tauró" que permeten nadar més eficientment, sabates de córrer de tecnologia punta, etc. Aquesta tecnologia fa ús d'eines com el podòmetre o superfícies d'equilibri, mitjans audiovisuals (vídeo, àudio), i interacció social (fer esport amb una altra persona).
4. Les **tecnologies de rehabilitació** són substàncies i procediments per tractar lesions. Els fàrmacs són administrats per especialistes en entrenament i medicina esportiva (anti-inflamatoris). Els mètodes inclouen màquines d'aigua en moviment giratori i equips d'ultra-so per a què els/les atletes puguin tractar els seus músculs i articulacions amb dolor. Més recent, les corrents d'estimulació elèctrica a la zona afectada indueixen el flux sanguini i ajuden en el procés de curació.
5. Les **tecnologies de moviment** fan referència als dispositius i procediments dissenyats per avaluar la forma i la eficiència del cos de l'atleta. La més comuna és l'anàlisi de vídeo, encara que hi ha moltes més eines sofisticades que aporten informació computeritzada i detallada en la biomecànica de l'atleta. Exemples com l'ús de la tecnologia de vídeo d'alta velocitat, de (mini) càmeres digitals, sensors en el cos, transmissió sense fil i mòbils-ordinador.
6. Les **tecnologies de bases de dades** inclouen innovacions informàtiques que permeten als atletes i entrenadors/es tenir coneixements dels rivals i

d'ells mateixos. Els programes de bases de dades afecten en la manera de fer la feina dels professionals, ja siguin entrenadors o esportistes.

Adicionalment, altres tecnologies que es poden emprar en el món de l'esport són les següents [23]:

- Tecnologies per al control de l'entrenament i la competició (pulsòmetres, analitzadors portàtils de gasos: VO_2 max, O_2 , CO_2 , VCO_2 , analitzadors portàtils de lactat).
- Tecnologies pel desenvolupament de les capacitats físiques (aparells generadors de hipòxia, plataformes vibratòries)
- Tecnologies aplicades a l'avaluació tàctica dels esports.
- Sistemes de seguiment de la mirada.

En definitiva, la tecnologia continua canviant la manera en què els esports es practiquen, com es tracten les lesions, i de quines maneres es milloren els resultats del rendiment esportiu.

2.1.1. Articles

A continuació, s'estudiaran unes fonts referents a la visualització de dades en l'àmbit esportiu. Donat que la tecnologia de l'esport abasta tot un món de possibilitats per al desenvolupament d'eines informàtiques, el que s'ha volgut en aquest cas, és analitzar quina és la manera més efectiva i òptima per a representar les dades i en quins casos s'ha trobat aquest problema i de quina manera s'ha solucionat.

2.1.1.1. Data-Driven Visual Performance Analysis in Soccer

A nivell d'elit, el personal tècnic necessita capturar, processar i analitzar grans quantitats de dades, a l'hora de mesurar el rendiment en els seus respectius

equips i oponents, com també avaluar perspectives potencials amb la fi d'aportar informació integrada ràpidament per propòsits d'entrenament [3].

La tecnologia actual utilitza sistemes de ràdio-freqüència, sistemes de visió computacional semi-automàtica, o unitats diferencials i no diferencials de GPS, les quals permeten capturar dades de les posicions dels jugadors, amb un nivell de precisió molt tolerable.

No obstant això, la seva utilitat aparent es veu fortament disminuïda per la manca de cap eina de programari estructurat capaç de comprendre fàcilment aquestes variables.

A l'analista se l'hi ha d'oferir un panell de comandaments interactiu que:

- a) Mostri mètriques del físic i els canvis de posició dels jugadors en una manera visual a l'usuari final.
- b) Habiliti un anàlisi espai-temps i multivariat del partit.
- c) Redueixi la càrrega intel·lectual (cognitiva) de la tasca d'anàlisi tot assegurant una interacció adequada amb l'ordinador.

Es podria fer ús de patrons de moviment i mètriques complexes de coordinació en els canvis de posició dels jugadors per explicar el comportament tàctic col·lectiu en partits de futbol.

2.1.1.1.1. Variables Utilitzades

En aquest article [3] s'utilitzen variables per a un anàlisi a nivell d'un equip esportiu col·lectiu, és a dir, de més d'un jugador; i per a un anàlisi individual utilitzant altres mecanismes d'anàlisi per a l'esport.

- Col·lectives: Variables per a analitzar les dades de forma grupal de l'equip en qüestió. El seu valor s'aconsegueix en funció de les dades recollides en conjunt de l'equip.
 - ❖ *Centroid*. [11] Aquesta variable captura moviments oscil·latoris en un grup de jugadors com els moviments en direcció a les línies de

gol o a les bandes o en direcció en contra, és a dir, les posicions i les àrees de superfície del *centroide* dos equips de futbol descriuen, potencialment el flux coordinat d'atac i defensa durant partits de futbol a petita escala. Es calcula a partir de la posició mitjana de totes les coordenades dels jugadors del camp en un moment determinat. Mesures utilitzades:

- Distància al *centroide*.
- Distància al *centroide* dels oponents.
- Distància entre els *centroide* dels equips.

❖ *Stretch Index*. [9] També conegut com el radi, representa el valor mitjà de dispersió dels jugadors al voltant del *centroide* de l'equip, i és un paràmetre per interpretar la dinàmica de les tàctiques de l'equip. Captura la relació sinèrgica contra-fase de comportaments de contracció i extensió d'equips com a funció d'intercanvi de la possessió de la pilota. Mesures utilitzades:

- Velocitat d'elongació o escurçament de la dispersió dels equips al camp.
- Espai de joc efectiu o àrea de superfície. Expressa la relació entre les formes i els espais ocupats en els dos equips.

❖ Sincronia d'equip. Estadística de processament de senyal que descriu la sincronització d'una manera no lineal, aportant una mesura quantitativa de coordinació entre dos oscil·ladors. S'utilitza per entrenaments tàctics per aconseguir moviments simètrics entre individus en certes situacions d'atac o defensa.

- L'algoritme *ApEn* pot ser utilitzat per identificar regularitat en patrons de moviment de jugadors [26].

- Individuals: Variables que es focalitzen en les dades recollides de cada jugador, així s'especifica de manera singular cada subjecte de l'anàlisi.
 - ❖ Comportament de l'equip. Resultat de molts treballs individuals en interacció, pot estar fet en un mode geo-espacial.

- Els mapes de calor suposen una manera útil i simple de visualitzar variables espai-temps com ara la posició o la velocitat.

S'han utilitzat mapes de calor [33], histogrames espacials, per representar gràficament variables espai-temps com la posició o la velocitat, tal i com es mostra en la figura 6 per aportar una millor imatge del rendiment del jugador al llarg del temps. Hi ha molts mapes de calor diferents que es poden desenvolupar en el futbol. Per exemple, es pot contemplar on es reben les passades.

Aquesta és una bona manera de valorar els davanter, que fan menys passades, però sovint es troben a les finals. També es poden realitzar mapes de calor de la defensa mirant on els jugadors recuperen la pilota.

La zona, del mapa de calor, més vermella o més “calenta” és on hi ha hagut més passades o on hi ha hagut o ha succeït el que es vol estudiar amb el mapa.

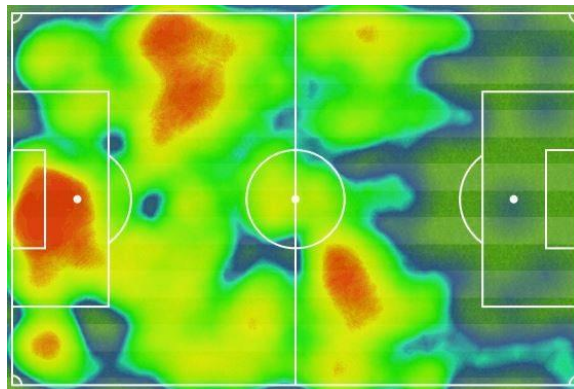


Figura 6. Exemple mapa de calor.

2.1.1.1.2. Flux de treball

Tal i com es mostra en la figura 7, hi ha dos modes principals d'operació per aquest flux de treball. En el primer, l'usuari comença amb una hipòtesi que vol validar, entra els paràmetres (interval de temps, velocitat de reproducció, selecció de jugadors), i els ajusta a la complexitat del problema, després en la secció d'anàlisi ha de escollir si vol fer un anàlisi individual (configurant els seus

mecanismes corresponents al tipus d'anàlisi), o un grupal (col·lectiu, amb les seves variables d'anàlisi).

Un cop acabat l'anàlisi, s'interpreten els resultats i, en aquesta part, arriba a la conclusió si ha de re-formular la seva hipòtesi inicial donats els resultats o si aquests són adequats, doncs donar la hipòtesi com a bona i extraure les dades resultants.

En el segon mode, l'analista explora les dades sense una hipòtesi prèviament coneguda i aquesta hipòtesi es forma un cop detectat un esdeveniment particular en les visualitzacions seguint els passos descrits anteriorment.

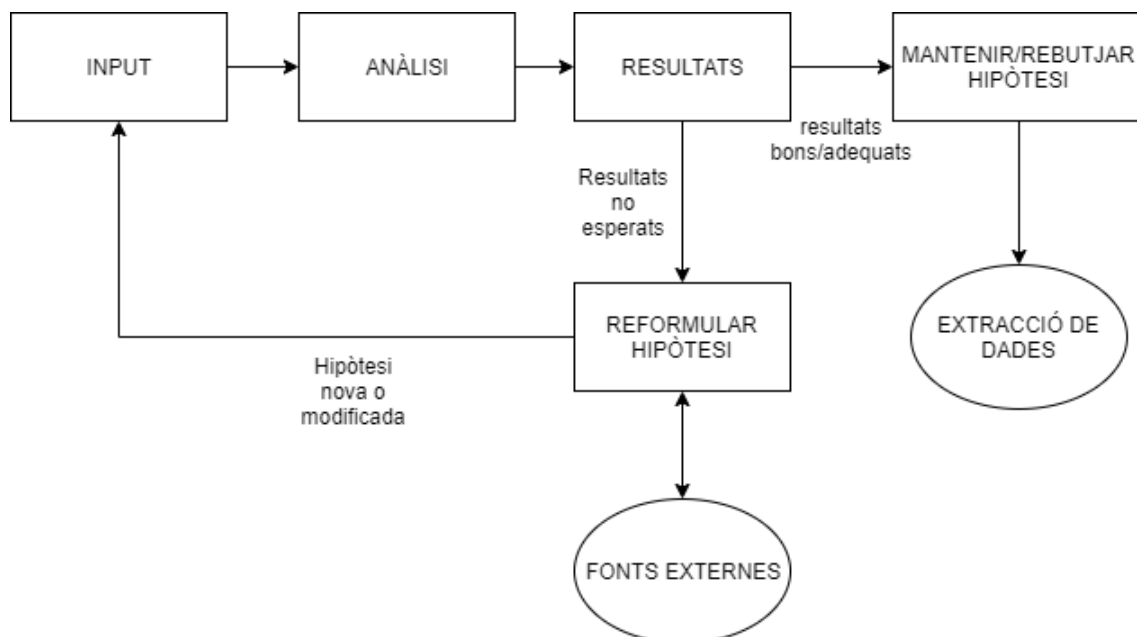


Figura 7. Flux de treball article anàlisi futbol.

En aquest estudi [3], es demostra que incloent tècniques d'anàlisi visuals en anàlisis de comportament d'esports col·lectius pot produir resultats d'alta qualitat i una experiència satisfactòria amb la computadora en el mateix moment. Es dissenya un complement iteratiu de flux de treball per aquesta eina d'anàlisi esportiu que captura i accelera, eficaçment, molts fluxos de treball experts que han de ser realitzats manualment o semi-automàticament en el passat.

S'ha trobat que l'aplicació de visualització de dades i les tècniques de disseny centrat en l'usuari ajuda a mantenir la corba d'aprenentatge general del sistema sense comprometre la precisió i l'eficiència de l'anàlisi.

2.1.1.2. Digital Analysis and Visualization of Swimming Motion

La investigació en simulacions computeritzades de flux ha millorat la forma d'entendre com l'impuls i la frenada poden ser optimitzats per un millor rendiment. No obstant, en el món de la natació, portar aquesta informació a la millora tècnica pot resultar difícil. En aquest article [18] es presenta un *framework* d'anàlisi i visualització pel moviment de la natació, el qual utilitza realitat virtual per a mostrar models tridimensionals de nadadors/es.

L'objectiu principal del projecte és proveir als usuaris finals un sistema que els permeti digitalitzar els seus moviments des de seqüències de vídeo, crear representacions virtuals personalitzades. A més a més que sigui capaç d'utilitzar informació sobre nadadors/es individuals per a transformar models prototip d'aquests/es, per l'anàlisi i comparació dels seus moviments amb els d'altres competidors emmagatzemats en una llibreria, utilitzant un mètode de visualització interactiva 3D. L'ús de la realitat virtual alleuja molts problemes associats als mètodes de visualització basada en vídeo actuals per analitzar el moviment de la natació.

En general, per analitzar el moviment, hi ha dos mètodes principals. El primer es basa en el vídeo-anàlisi, comparant els vídeos dels atletes principiants amb els dels d'elit. Amb els fragments del vídeo, primer s'ajusta manualment el model humà 3D al subjecte en el primer *frame* del vídeo, després una sostracció del fons, una extracció de la silueta i un filtratge de *Kalman* [2] discret (solució òptima de tasques de *tracking* i de predicció de dades. Utilitzat en anàlisis de moviment visual.) per predir la postura del subjecte en cada *frame*.

El desavantatge més comú en sistemes basats en vídeo és la seva limitació de l'angle de la càmera. En la natació, el vídeo és més problemàtic donats efectes òptics, com poden ser la reflexió, la refracció i efectes de l'aigua com esquitxos i bombolles. El mètode alternatiu es basa en la realitat virtual, on l'atleta pot aprendre i millorar el seu rendiment principalment a través de la interacció amb l'entorn virtual.

El projecte segueix el mètode de la realitat virtual. Encara que s'ha centrat en la visualització del moviment de la natació més que en el propi flux, la visualització d'aquest es pot incorporar fàcilment en el *framework*, gràcies a l'ús de mètodes de representació estàndards de *OpenGL*.

2.1.1.2.1. Captura de moviment

A través d'un mòdul d'emmagatzemament de mobilitat, el moviment digitalitzat es pot guardar en bases de dades *online* i pot ser usat per analitzar i visualitzar nedadors/es en 3D.

El mòdul aporta un sistema semi-automàtic per l'usuari per a marcar la localització de les extremitats en un nombre de *frames* en el vídeo.

Es col·loca una figura articulada prototípica en la part més alta del *frame* del vídeo, permetent a l'usuari veure les posicions de les extremitats. L'usuari clica primer en les posicions de totes les articulacions de la imatge de cada *frame* del vídeo. Mitjançant aquestes posicions, es calcula la longitud de cada os. L'orientació de cada parell d'ossos es calcula a partir de les posicions articulars, així com l'angle entre ells.

El mòdul genera com a resultat dades de mobilitat 3D que corresponen al vídeo. Aquestes dades es poden guardar utilitzant el mòdul d'emmagatzemament i es



Figura 8. Exemple captura digital de moviment.

poden exposar amb el mòdul d'anàlisi i visualització tal i com es mostra en la **figura 8**.

2.1.1.2.2. Parametrització

Adicionalment, la parametrització del moviment (formes del cos humà) requereix un *mapping* de mesures antropomètriques (exemple: longitud espatlla-colze) en els models resultants.

En el projecte s'usa mesures antropomètriques estandarditzades. Tres capes de control diferents aporten un procediment per general varies formes amb l'input de:

- i. Alçada i pes.
- ii. Pes i longitud del ossos (pes, alçada, amplada de les espatlles, amplada dels trocànters del fèmur, longitud de l'avantbraç, longitud del peu, etc.).
- iii. Totes les mesures.

S'inclou tres mesures:

- i. Distàncies d'Euclides entre dos punts de referència.
- ii. Distàncies axials entre dos punts de referència respecte un eix.
- iii. Distàncies tangents (dos punts de referència sobre la superfície de la pell).

Les mesures antropomètriques són "mapejades" en un conjunt de restriccions lineals entre 114 punts de referència, les quals són característiques ben definides sobre la pell, respectant les estructures òssies.

2.1.1.2.2. Visualització

Amb un plantejament de visualització interactiva 3D, l'usuari té un control il·limitat sobre l'angle de visió. Aquest control facilita la rotació i *zoom* lliurement, presentant múltiples vistes complementàries consegüentment. Aquestes són

sincronitzades i ajuden a entendre moviments complexes, mostrant possibles regions d'oclusió.

Depenent del que requereixi l'usuari, l'animació es pot presentar de maneres diferents:

- Ús de representació sòlida amb ombrejat.
- Ús de mode de representació de vèrtexs per mostrar l'esquelet, com també el/la nadador/a.
- Ús d'un mode de representació de silueta per emfatitzar el canvi en els angles de les articulacions, tal i com es mostra en la **figura 9**.

La segona eina d'anàlisi del sistema visual proporciona monitoritzar traces (rastres) de les posicions de les articulacions. D'aquesta manera, l'usuari pot observar com les localitzacions de les articulacions canvien durant el temps, i així poden fer millores necessàries en les seves tècniques.

Apart dels mètodes de presentació d'informació que proposa el sistema general d'aquest projecte, es pot visualitzar simultàniament múltiples nadadors/es. En la funcionalitat comparativa, dos conjunts de dades de moviment poden ser carregades des de la llibreria de mobilitat. Els usuaris poden realitzar qualsevol combinació visual.



Figura 9. Exemple tracking format silueta.

La meta d'aquest projecte d'investigació, és que el sistema que s'ha proposat l'utilitzin atletes i entrenadors/es de qualsevol categoria. Per tant, s'ha assegurat la compatibilitat entre una gran varietat de solucions de software.

Encara que el sistema de visualització s'ha dissenyat per la natació, és prou general per poder ser aplicat en una àmplia gamma d'esports, on l'anàlisi de canvis tènues en una mobilitat cíclica o repetitiva és important. A més a més, es pot incorporar informació del flux en el *framework* de visualització.

2.2 Estudi dels fitxers de dades a gestionar

S'ha treballat en fitxers de dades extrets de la pràctica del futbol. Segons la captura de dades es pot diferenciar dos tipus de fitxers; un fitxer d'esdeveniments durant la pràctica esportiva i un fitxer o més d'un de dades de dispositius addicionals de rastreig (acceleròmetre, GPS, etc.). Aquestes dades s'han extret a partir d'equipament específic i en un entorn controlat durant una classe magistral en el INEFC de Lleida. Remarcant que cadascun dels fitxers a tractar tenen una freqüència de mostreig diferent menys el d'esdeveniments el qual no en té.

Apart de la freqüència, la característica més distingida i la que pot provocar conflictes o problemes a l'hora de tractar els fitxers és el factor temps, és a dir, el format en el que està especificat el temps en què s'han capturat les dades de les variables de rendiment. Mentre que el fitxer d'esdeveniments té una columna de *frames*, una de temps en segons i una altra de temps en mili-segons, l'altre mena de fitxer només té una columna de temps el qual no és igual a cap dels del fitxer d'esdeveniments.

Per tant, s'ha de trobar una manera per a què hi hagi una concordança entre ambdós tipus de fitxers per poder correspondre totes les dades en els mateixos instants de temps i, en conseqüència poder-les representar i analitzar a la vegada.

Aquesta correspondència de dades de diferents fitxers també vindrà donada amb la equiparació de les seves freqüències gràcies als mètodes de transformació de format estadístic de dades: mètode "interpolació" (canviar d'una freqüència

menor a una major) i mètode “*downsampling*” (canviar d’una freqüència major a una menor).

2.3. Selecció de la plataforma de desenvolupament i desplegament del projecte

En aquest apartat, s’exposaran les tecnologies que s’han estudiat per determinar quina serà la plataforma per desenvolupar el projecte.

2.3.1. Llenguatge R i Shiny

R és un entorn de programació, anàlisi estadístic i gràfic, derivat del llenguatge S i distribuït amb llicència GNU, per tant, lliure. Està constituït per més de 1.400 paquets integrats amb els que es fa possible executar simples anàlisis descriptius o aplicar complexos models formals. Interfícies gràfiques com *Rcommander* que crea entorns de treball similars al SPSS (*Statistical Package for the Social Sciences*) [12].

El sistema R disposa de funcions bàsiques pels anàlisis descriptius més elementals (mitja aritmètica, desviació estàndard, variància ...) i pels models formals més complexos derivats dels últims avanços en el camp de l’estadística, psicometria, mineria de dades, econometria i finances, ecologia, màrqueting, estadística robusta, estadística espacial, entre d’altres [24].

Shiny és un *framework* per la construcció d’aplicacions web interactives utilitzant codi R [22]. La vinculació automàtica i “reactiva” entre els inputs i els outputs i *widgets* (components gràfics, o controls, amb el qual l’usuari interacciona, com per exemple, una finestra, una barra d’eines o una caixa de text.) prefabricats permeten construir aplicacions elegants, *responsive*¹, i potents amb un esforç

¹ tècnica de disseny web que busca la correcta visualització d’una mateixa pàgina en diferents dispositius

mínim. Es poden crear aplicacions Shiny sense tenir coneixements de HTML, CSS, o JavaScript.

Aquest *framework* està creat a partir de blocs de construcció que tenen uns principis robustos d'enginyeria de software.

La seva interfície d'usuari pot ser personalitzada o estesa fàcilment, i el seu servidor usa programació reactiva [15], la qual ajuda a crear qualsevol lògica de *backend*. Aquesta programació és un paradigma enfocat en el treball amb fluxos de dades finites o infinites de manera asíncrona, a més a més de ser compatible amb altres paradigmes que magnifiquen les seves característiques. Aquest paradigma té dos objectius principals:

1. Propagar els canvis en un sistema requerint l'esforç mínim.
2. Treballar més fàcilment amb fluxos asíncrons de dades.

Donat que del llenguatge R no es coneix en profunditat i per això, a la pràctica es desconeixen moltes utilitats i tampoc s'ha treballat en projectes grans durant el Grau, no proporciona la confiança necessària per a desenvolupar el projecte amb aquesta tecnologia.

2.3.2. Node.js (*back-end*)

Node.js és un entorn d'execució multi-plataforma [28], de codi obert, desenvolupat i per a desenvolupar amb JavaScript, orientat a esdeveniments asíncrons i llicenciat amb la X11, variant de la llicència MIT, i una *Apache 2.0*.

Node.js utilitza un model d'entrada i sortida sense bloquejar basat en esdeveniments que el fa lleuger i eficient, perfecte per a aplicacions en temps real intensives en dades que es publiquen en dispositius distribuïts. A més a més, aporta una llibreria de variis mòduls de JavaScript.

Va ser dissenyat amb l'objectiu de crear programes de xarxa altament escalables, per exemple, aplicacions per la capa de servidor web. Una de les

motivacions del projecte va ser augmentar la capacitat de suportar una gran quantitat de connexions concurrents. Dins de *Node* el codi s'organitza per mòduls o paquets.

Per tant, Node.js ofereix accés a variïis d'aquests paquets junt amb els mòduls que el componen per a què qualsevol pugui utilitzar-los amb la fi de desenvolupar fàcilment la part del servidor d'una web. Gràcies al *Node Package Manager*(NPM) es realitza la gestió d'aquests paquets i aquests mòduls de la manera més eficient i ràpida possible, sense necessitat de preocupar-se de les dependències del projecte i la instal·lació de paquets.

NPM abasta un conjunt de components reutilitzables disponibles públicament a través d'un repositori en línia, amb la versió i la dependència de gestió. Qualsevol pot publicar el seu propi mòdul que serà inclòs en el repositori de NPM. Alguns mòduls més populars són: Express.js, Connect (servidor HTTP), socket.io i sockjs, jade (motor de plantilles), mongo i mongojs, coffee-script, entre d'altres.

Donat que de l'entorn Node.js no s'ha es desconeixen els mecanismes per començar i desenvolupar un projecte com el corresponent i per això es tardaria un temps a familiaritzar-se amb les seves utilitats no hi ha la necessitat per a desenvolupar el projecte amb aquesta tecnologia.

2.3.3. Angular (*front-end*)

Angular [4] és un framework de desenvolupament per a JavaScript creat per Google. La seva finalitat és facilitar el desenvolupament d'aplicacions *Single Page App*(SPA) i donar un conjunt d'eines per treballar amb els elements d'una web d'una manera senzilla i òptima.

Una aplicació web SPA creada amb Angular és una web d'una sola pàgina, és a dir, la navegació entre seccions i pàgines de l'aplicació, així com la càrrega de dades, és realitza de manera dinàmica, asíncronament amb el servidor i sense refrescar la pàgina. Per tant, les aplicacions en Angular són reactives i no

recarreguen el navegador gràcies al dinamisme i sincronisme d'AJAX (*Asynchronous JavaScript and XML*) [27].

El seu objectiu principal és separar completament el *front-end* i el *back-end* d'una aplicació web. Una gran potència és la possibilitat d'utilitzar plantilles declaratives(en HTML) per definir la interfície, aplicar injeccions de dependències i crear components reutilitzables.

Un dels avantatges d'Angular és que utilitza el llenguatge *TypeScript* per a la lògica de negoci, el qual és de fàcil manteniment i està suportat amb una documentació consistent.

Malgrat s'hagi treballat en una assignatura del Grau, no transmet a l'autor la suficiència per abordar el problema del projecte amb aquesta tecnologia.

CAPÍTOL III

Anàlisi i disseny del projecte

En aquest capítol s'exposaran les diferents parts de l'anàlisi pel posterior disseny de l'aplicació i la seva implementació. Es veuran les parts del blast-off del projecte, els casos d'ús de l'aplicació amb els seus respectius casos com a producte software i les user-stories per a cada funcionalitat trobada. Addicionalment un apartat per mostrar els requeriments refinats, el qual s'ha hagut de dur a terme donades les circumstàncies al llarg del projecte.

3.1. *Blast-off* del Projecte

El *blast-off* identifica el límit de l'àrea de treball on el producte formarà part, i determina el propòsit que el producte ha de completar. Durant aquesta activitat és quan es decideix els factors clau que determinen la viabilitat dels requeriments del Projecte.

L'activitat *blast-off* recull suficient informació per assegurar que el Projecte pot prosseguir fluidament. També verifica que el Projecte és viable i mereix la pena l'esforç. La majoria dels resultats serveixen com a fonament per l'activitat de rastreig de requisits que esdevindrà.

El treball és la part de l'organització que es necessita estudiar per descobrir els requeriments. Normalment està connectat a altres parts de l'organització i al món exterior. S'ha d'estudiar el treball suficientment bé per entendre com funciona. Aquest enteniment possibilitarà arribar a abasts alternatius pel producte i finalment triar l'únic per construir.

La conclusió: Primer entendre el treball, llavors decidir quin producte proporciona el millor valor en aquest treball.

En aquest TFG, gràcies al *blast-off*, es produiran les entregues següents, entre altres:

1. El propòsit del Projecte: la raó per fer la inversió en la construcció del producte i l'avantatge pel negoci que es vol aconseguir amb la solució.
2. L'abast del treball: l'àrea de negoci o domini a estudiar.
3. Els implicats: les persones amb un interès o amb alguna influència en el producte.
4. Limitacions: restriccions de l'abast o de l'estil del producte.
5. Noms: terminologia utilitzada en el Projecte.
6. Fets rellevants i suposicions: influències externes que poden fer diferent el producte, o supòsits fets pels desenvolupadors.

3.1.1. Directrius del Projecte – raons i motivacions pel Projecte

3.1.1.1. El propòsit del Projecte

La raó per fer la inversió en la construcció del producte és automatitzar l'anàlisi que es duu a terme en el rendiment de diferents esports. Aquest anàlisi comportaria dos objectius principals: el tractament que es vol de les dades derivades i la visualització gràfica i no gràfica de les variables de rendiment. Això suposa implementar una plataforma o aplicació web la qual millori i mostri l'anàlisi esmentat juntament amb la capacitat de que pugui ser flexibilitzat (modificat en temps real) per l'usuari final.

L'avantatge pel negoci de l'esport que es vol aconseguir amb la solució de software que es planteja en el Projecte, és doncs estudiar el comportament tan col·lectiu com individual, segons l'esport, i els patrons de moviment per tal així guanyar avantatge esportiu en el món professional o no professional.

Per garantir el primer gran objectiu, es proposa gestionar les dades de manera eficient en un aspecte computacional. L'avantatge d'aquesta proposta seria reduir el temps de còmput a l'hora de realitzar les gestions pertinents per l'usuari i d'aquesta manera que hi hagués un temps de resposta mínim i acceptable. Aquest objectiu es complirà de tal forma que el processament d'aquestes dades no ocupi més del 15 o 20% de temps total que l'usuari vulgui dedicar a l'estudi o anàlisi del rendiment de l'esport desitjat.

Respecte a garantir el segon gran objectiu, el que es proposa és oferir una manera simple però efectiva per a l'usuari de com es pugui observar al màxim detall tota la informació que s'hagi volgut manipular. Es podrà observar i interactuar amb totes les dades d'una manera molt intuïtiva i còmode. Per aconseguir-ho, el 99% de l'anàlisi de les dades es podrà visualitzar sense cap error.

A part d'aquests objectius primordials, també és important posar èmfasi en el tema d'integrar les dades, capturades prèviament, a l'aplicació. És per això que un altre objectiu serà integrar aquestes dades en la pròpia aplicació. Una bona proposta per aquesta finalitat seria un mode d'integració de dades eficient i de ràpida resposta de processament. D'aquesta manera s'evitaria que s'incrementés el temps d'espera en tractar les dades i poder visualitzar l'estudi final durant la sessió de l'usuari.

3.1.1.2. Implicats en el producte

- El patrocinador paga pel desenvolupament del producte. Pot ser: departament de màrqueting, desenvolupament del producte, gestió d'usuaris.
- El client compra el producte. Es necessari conèixer prou bé aquesta persona per entendre el que troba valuós i, per tant, el que comprarà.
- Usuaris. El propòsit d'identificar els usuaris és entendre què és el que estan fent i quines millores consideren de valor.

En aquest Treball el patrocinador és l'autor del Treball, el client és el co-director Àngel Ric, i els usuaris són tots els professionals en l'àmbit de l'esport.

3.1.1.3. Usuaris del producte

Aquest producte el podran utilitzar les persones que vulguin o hagin de fer un estudi exhaustiu del seu rendiment o del rendiment del seu equip. Llavors, el producte ha de ser simple d'utilitzar, entenedor i que estigui molt relacionat amb termes del món de l'esport, per a què aquestes persones es sentin ben acollides i puguin realitzar un bon anàlisi esportiu.

Per tant com a usuaris principals hi ha:

- Esportistes professionals
- Analistes/Científics esportius
- Persones esportistes en un àmbit no professional

3.1.2. Restriccions del projecte i del producte

3.1.2.1. Restriccions als requisits

Pel que fa a les restriccions del disseny, s'ha de tenir en compte que el Projecte el limitarà la tecnologia que s'utilitzi per desenvolupar el Projecte.

En aquest cas només es projectarà la plataforma via ordinador. No obstant, el que realment es vol es que no hi hagi cap problema en poder iniciar sessió de la plataforma web en qualsevol màquina computacional (ordinador, màquina virtual).

3.1.2.2. Acords de noms i definicions

Seguidament, es definirà tot el vocabulari per ajudar tan al lector per una lectura fluida i entenedora com a l'autor per conèixer i saber en tot moment tot el relacionat amb el projecte i el seu desenvolupament.

Patró de moviment: [7] Conjunt d'accions o moviments organitzats en seqüència ordenada, la qual permet exercir una funció motriu òptima del cos humà. Exemples com la flexió, la extensió, etc.

Rendiment esportiu: [5] El rendiment esportiu és la manera en la qual la participació esportiva és mesurada. És una combinació complexa entre funcions biomecàniques, factors emocionals i tècniques d'entrenament. Té una popular implicació a l'hora de representar el percaç de l'excel·lència, per tant, es pot dir que és una progressió cap a l'excel·lència o els objectius marcats.

Variable de rendiment: Mostra relació entre dos components d'un sistema individual o col·lectiu.

Un equip de qualsevol esport seria un sistema col·lectiu i els seus jugadors els components, en aquest cas, una variable de rendiment podria ser la distància de relació entre dos o més components (jugadors).

Un/a esportista seria un sistema individual (l'aparell locomotor) i el seu moviment o patró de moviment a examinar podria ser la flexió del colze, en aquest cas, els seus components serien l'espatlla, el colze, el canell i la variable de rendiment seria l'angle del colze en fer el moviment.

Dada posicional: Conjunt de dades que permet identificar la posició d'un component de qualsevol objecte o subjecte susceptible a ser capturat per un sistema de tracking i fer un seguiment de la seva posició espacio-temporal.

Sistema de tracking: [6] Sistema capacitat de representar espai virtual al observador (humà) mentre segueix (rastreja) les seves coordenades.

Proporciona una seqüència ordenada de dades d'ubicació per a processaments posteriors.

GPS: [36] Sistema de posició i/o de navegació per satèl·lit que permet conèixer amb precisió la situació geogràfica i temporal en qualsevol lloc de la Terra. Permet determinar la posició de qualsevol objecte o subjecte connectat en aquest sistema.

UWB: [17] La tecnologia *Ultra-Wide-Band* combinada amb múltiples antenes de transmissió i recepció és una manera viable d'aconseguir taxes de dades superiors a 1 Gb/s per a comunicacions sense fil. S'aplica normalment a comunicacions de curt abast i, per tant, principalment a l'interior en entorns caracteritzats generalment per una densa propagació de múltiples rutes.

Acceleròmetre: [10] Dispositiu que mesura l'acceleració múltiples vegades dins una freqüència donada i ho resumeixen com un recompte durant un període de temps pre-definit. El recompte resultant representa una acceleració dins el període de temps. Els acceleròmetres eliminen els prejudicis associats a les mesures d'autoreportatge de l'activitat física.

Giroscopi: [21] Dispositiu que mesura o manté el moviment de rotació. Sistemes microelectromecànics (MEMS) són petits sensors dins el giroscopi per mesurar la velocitat angular.

Freqüència de mostreig: Quantitat de mostres que es prenen per segon d'una senyal contínua (analògica) per a convertir-la en una digital. S'expressa en hertz o en cicles per segon.

Interpolació: Obtenció de noves dades partint d'haver conegut un conjunt inicial de dades. Per això s'augmenta la freqüència de mostreig del conjunt inicial, per tant el conjunt resultant és més voluminós que l'inicial.

Downsampling: Dividir una quantitat d'informació entre un enter o una fracció racional. Exemple pràctic: obtenir un àudio a 50.000Hz i convertir-lo a 27.500 Hz per emetre'l per un altre tipus de canal, com ara per ràdio.

3.1.2.3. Fets rellevants i suposicions

Com a suposicions, cal tenir en compte diferents aspectes referents a funcionalitats diferents d'altres plataformes amb el mateix objectiu final. Algunes d'aquestes poden ser, per exemple, les tècniques de visualització de les dades. Encara que la influència externa que pot fer diferent el producte sigui la pròpia creativitat i base d'idees del desenvolupador de la plataforma.

3.1.3. Requisits Funcionals – la funcionalitat del producte

3.1.3.1. L'abast del treball

El domini a estudiar engloba tots els aspectes del rendiment esportiu, és a dir, la condició física de l'esportista del qual s'analitza el comportament, l'esport que practica, l'àrea d'entrenament d'on es poden extreure les dades a partir dels seus exercicis físics.

3.1.3.2. L'abast del producte

L'abast del producte seria a nivell local, ja que és fruit d'un treball d'investigació d'universitat.

El producte tindrà una connexió directa amb els dispositius que facilitin les dades capturades per poder tractar-les.

3.1.4. Requisits No Funcionals – les qualitats del producte

3.1.4.1. Requisits d'usabilitat

A continuació es donarà la llista de requeriments per a la usabilitat de l'aplicació per a què l'usuari es senti còmode i no hagi d'esforçar-se molt en navegar per la web.

- La interfície ha de ser intuïtiva i simple de fer anar, proporcionant un alt nivell d'interactivitat i usabilitat.
- Alt nivell de navegació i presentació.
- El temps d'aprenentatge i l'esforç que ha de fer l'usuari en utilitzar per primer cop l'aplicació ha d'ésser mínim.
- El feedback ha d'acollir el confort de l'usuari. L'usuari s'ha de sentir familiar en l'ús de l'aplicació en tot moment.
- L'aplicació ha d'acomodar-se a canvis. Ha de tenir prou flexibilitat.

3.1.4.2. Requisits de rendiment

Seguidament, es llistarà uns requeriments aproximats per garantir un bon rendiment de processament de l'aplicació software.

- El temps en mostrar el tractament aplicat en les dades introduïdes a l'aplicació, és a dir, la visualització de l'anàlisi indicat serà màxim entre 5 i 10 segons.
- El temps en integrar les dades prèviament capturades serà menys de 3 segons.
- El temps en iniciar-se l'aplicació serà menys de 2 segons.

3.1.4.3. Requisits legals

La gestió de dades s'ha d'ajustar als requisits de la Llei Orgànica de Protecció de Dades (LOPD) i al reglament europeu de protecció de dades (RGPD).

3.2. Business UseCases, ProductUseCases, *UserStories*

S'han utilitzat Escenaris per modelar i registrar els casos d'ús de negoci (Business UseCase, BUCs), perquè l'escenari explica la història d'un BUC. Normalment és més ràpid descobrir la funcionalitat requerida quan es treballa amb escenaris que quan es codifica prototips [31]. **Un Escenari és un guió d'una trama, o una seqüència postulada de passos.**

El BUC sempre és una col·lecció de processos identificables, dades que són recuperades i/o emmagatzemades, resultats generats, missatges enviats, o alguna combinació d'aquestes. El BUC és una unitat de funcionalitat. Aquesta unitat és el fonament d'escriure els requeriments funcionals i no funcionals. El cas d'ús de negoci és la unitat de treball per estudiar més pràctica [31].

El PUC (*ProductUseCase*) és aquella part del cas d'ús de negoci pel qual es decideix construir un producte. El cas d'ús de producte precisa la funcionalitat del producte.

Les *UserStories* són una manera de representar els requeriments. La forma usual és: **Com a [rol], vull (característica) per així [raó]**

Una *UserStory* descriu una funcionalitat que serà valuosa tan per un usuari com un comprador d'un sistema o un software. Les *UserStories* es componen per tres aspectes:

- Una descripció escrita de la història utilitzada per a la planificació i com a recordatori.

- Converses sobre la història que serveixen per exposar els detalls d'aquesta.
- Tests que expressen i documenten detalls i que poden ser utilitzats per determinar quan una història és completa.

Per finalitzar l'etapa s'han construït uns prototips de les funcionalitats trobades a partir del rastreig, recerca de requisits. Tanmateix, són creats per tenir una visió general de com es mostren aquestes funcionalitats, en cap moment es proclamen com a maqueta definitiva per a la fase d'implementació i disseny.

3.2.1. The Brown Cow Model

A continuació, es mostrarà les característiques i les perspectives que s'ha seguit del model Brown Cow que es mostra en la **Figura 10**.

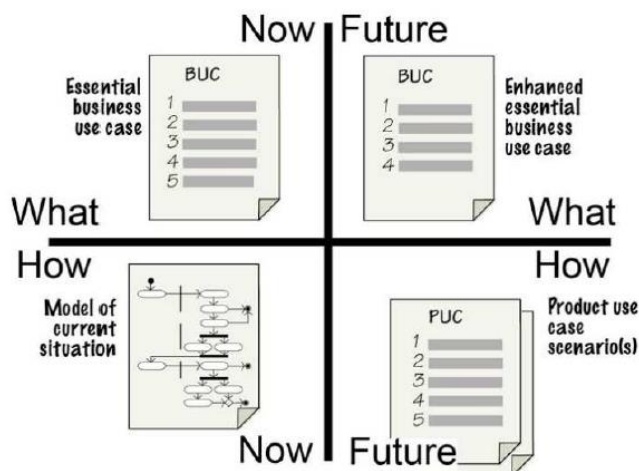


Figura 10. Brown Cow Model.

Es mostra quatre visions del treball, cada una aporta informació, a l'analista de negoci i als implicats, que és útil a diferents etapes del procés del descobriment de requeriments.

És convenient separar la essència del problema de qualsevol solució proposada. L'essència existeix independentment de qualsevol implementació tecnològica.

En aquest Treball s'ha desenvolupat les visions *What-Now* i *Future-How* del *The Brown Cow Model* per l'anàlisi i cerca de requeriments [31].

- *What-Now* mostra el principi real de negoci, o com es prefereix anomenar-ho, la essència del treball. Aquesta visió és completament i tecnològicament neutral, i mostra el negoci tal com existeix sense màquines, gent, o departaments d'organització. S'utilitza aquesta visió per netejar les idees sobre el que el negoci actual està realment realitzant sense inhibir-lo fent referència a processadors i artefactes físics que podrien no ser part de la futura implementació.
- *Future-How* mostra la visió idealitzada del principi de negoci de futur, i l'incrementa o millora amb la tecnologia i la gent necessària per portar-lo al món real.

3.2.1.1. BUCs i PUCs

Aquí es presentaran els casos d'ús més essencials per a la futura aplicació, tan els necessaris per al negoci (els BUCs) com els per a construir el producte (els PUCs)

3.2.1.1.1. Selecció variables de rendiment

Les dades capturades i enregistrades prèviament es divideixen en diferents parts segons la captura. Aquesta divisió s'organitza en diferents variables de rendiment. L'usuari analista esportiu necessita seleccionar les variables de rendiment adients al seu anàlisi esportiu.

What-Now

El cas d'ús de negoci sense haver implicació de cap tipus de software. Que ha de fer l'analista per seleccionar les variables, quines condicions prèvies hi ha d'haver per la casuística.

BUC: Seleccionar les variables de rendiment.

TRIGGER: Realització d'un anàlisi de rendiment.

PRECONDICIÓ: Que hi hagi constància de les variables que es volen analitzar.

IMPLICATS: Analista esportiu.

Acció 1: L'analista vol executar un nou anàlisi de comportament esportiu.

Acció 2: L'analista examina tot el llistat de les variables de rendiment de què disposa.

Acció 3: L'analista selecciona les variables que vol tractar.

E3.1: L'analista vol seleccionar una variable que no es troba disponible dins el llistat.

Future-How

El cas d'ús de negoci considerant implicació d'un software operatiu. Com l'analista seleccionarà les variables des del software hipotètic i implicat, quines condicions prèvies hi ha d'haver per la casuística considerada.

PUC: Enregistrar les variables de rendiment que es volen seleccionar.

TRIGGER: L'analista selecciona les variables a visualitzar.

PRECONDICIÓ: Hi ha d'haver variables disponibles i que es puguin seleccionar per enregistrar-les.

IMPLICATS: Analista esportiu.

Acció 1: El producte mostra el llistat de variables disponibles.

Acció 2: El producte demana seleccionar les variables, de les quals es vol realitzar l'anàlisi i representació, i acceptar la selecció.

Acció 3: El producte enregistra la selecció de variables.

Acció 4: El producte tracta aquestes variables per la seva posterior visualització.

OUTCOME: Variables de rendiment enregistrades i tractades correctament per la representació posterior.

3.2.1.1.2. Filtratge de variables de rendiment

L'usuari analista esportiu necessita filtrar les dades de les variables de rendiment, prèviament interpolades a la màxima freqüència, segons un període de temps i una freqüència de mostreig prèviament definits, amb l'objectiu d'aprofundir l'anàlisi dels patrons de moviment, i en definitiva el rendiment de l'atleta del qual s'analitza la seva pràctica esportiva.

What-Now

El cas d'ús de negoci sense haver implicació de cap tipus de software. Quina seqüència de passos ha de realitzar l'analista per filtrar les variables de rendiment, quines condicions prèvies hi ha d'haver per la casuística.

BUC: Tria de dades de les variables de rendiment.

TRIGGER: Dades capturades en un temps i freqüència concrets per cada variable de rendiment.

PRECONDICIÓ: Que hi hagin dades de variables capturades i un marge de temps i una sèrie de freqüències a escollir pel posterior anàlisi.

IMPLICATS: Analista esportiu.

Acció 1: L'analista estableix un rang temporal i una freqüència determinada com a paràmetres generals durant tot l'anàlisi de dades.

Acció 2: L'analista tria les dades capturades segons un segon temps d'inici i un segon temps final rang de temps i una segona freqüència dins del mostreig de dades per a cada variable.

Acció 3: L'analista es prepara per examinar i interpretar les dades filtrades.

Future-How

El cas d'ús de negoci considerant implicació d'un software operatiu. Com l'analista filtrarà les variables de rendiment des del software hipotètic i implicat, quines condicions prèvies hi ha d'haver per la casuística considerada.

PUC: Filtrar i enregistrar les dades.

TRIGGER: Entrada de dades amb paràmetres de temps i freqüència concrets per cada variable.

PRECONDICIÓ: Que hi hagin dades de les variables capturades i que els temps i freqüències per l'anàlisi global establerts estiguin sincronitzats amb els de cada variable.

IMPLICATS: Analista esportiu.

Acció 1: El producte demana quin format de sèrie temporal es vol utilitzar en la sessió d'anàlisi actual, l'inici i el final de la sèrie temporal durant tota la sessió i una freqüència concreta.

A1.1: Sèrie temporal en *frames*.

A1.2: Sèrie temporal en temps físic real.

Acció 2: El producte demana introduir fitxers de dades de variables de rendiment, juntament amb un inici i un final de la captura de les dades guardada en cada fitxer i una freqüència determinada.

A2.1: Si la sèrie temporal utilitzada és en *frames*, el producte demanarà un nombre de *frames* en la introducció del rang temporal de cada fitxer.

A2.2: Si la sèrie temporal utilitzada és en temps real, el producte demanarà la introducció del rang temporal en un format de temps real (hora-minut-segon-dècima).

Acció 3: El producte mostra les dades segons el filtratge temporal i freqüencial establert.

OUTCOME: Rang temporal i freqüencial registrat per ser utilitzat durant la sessió d'anàlisi; dades de les variables mostrades segons rang temporal i freqüència de mostreig establerts.

3.2.1.1.3. Anàlisi i visualització de les variables de rendiment.

Amb les dades filtrades segons els paràmetres previs, estan preparades per ser visualitzades en format taula de dades per comoditat a l'usuari analista esportiu. D'aquesta manera l'analista podrà desenvolupar un anàlisi exhaustiu i trobar els patrons de moviment i de comportament de la pràctica esportiva objectiu, i guardar els resultats obtinguts.

What-Now

El cas d'ús de negoci sense haver implicació de cap tipus de software. Que ha de fer l'analista per contemplar i estudiar les dades corresponents, quines condicions prèvies hi ha d'haver per la casuística.

BUC: Representar i estudiar les dades filtrades segons la sèrie temporal prèviament definida.

TRIGGER: Ajustament de les dades segons els paràmetres globals de la sessió d'anàlisi amb els paràmetres de cada fitxer de dades, tan els temporals com els freqüencials.

PRECONDICIÓ: Que hi hagin dades, reals, interpolades o en *downsampling*, de variables de rendiment capturades dins el marge de temps i en la freqüència escollits.

IMPLICATS: Analista esportiu.

Acció 1: L'analista estudia les dades de les variables les quals l'hi convenen més segons la seva investigació.

Acció 2: L'analista separa les dades segons un paràmetre amb una condició per obtenir les dades de la variable a estudiar d'acord amb el seu variant criteri que segueix durant la sessió d'anàlisi.

E2.1: La condició que defineix sobre el paràmetre no l'hi proporciona cap dada capturada de la variable de rendiment tractada.

Acció 3: L'analista anota en una taula els resultats aconseguits gràcies al segon filtratge que realitza a les dades de les variables de rendiment que vol tractar durant la sessió d'anàlisi.

Future-How

El cas d'ús de negoci considerant implicació d'un software operatiu. Com l'analista visualitzarà les dades de les variables de rendiment des del software hipotètic i implicat, quines condicions prèvies hi ha d'haver per la casuística considerada.

PUC: Visualitzar i filtrar cada variable de rendiment segons l'estudi que es vol realitzar.

TRIGGER: Entrada de dades amb paràmetres de temps i freqüència concrets per cada variable.

PRECONDICIÓ: Que hi hagin dades de les variables capturades i que els temps i freqüències per l'anàlisi global establerts estiguin sincronitzats amb els de cada variable.

IMPLICATS: Analista esportiu.

Acció 1: El producte mostra en una taula, inicialment, totes les dades de totes les variables obtingudes dels fitxers d'entrada (fitxers .csv) dins la sincronització de la sèrie temporal i de la freqüència definida prèviament.

Cada columna presenta una de les variables seleccionades prèviament i cada fila la dada de la variable en un temps o *frame* concret.

Acció 2: El producte demana per cada columna/variable si es vol filtrar segons una condició concreta.

E2.1: La condició no pot ser complida per les dades capturades d'aquella variable.

E2.2: El producte mostra les files en blanc.

Acció 3: El producte ofereix una opció de guardar els resultats del segon filtratge que l'analista ha aplicat en cada variable de rendiment en un format xls o csv.

OUTCOME: Visualització del segon filtratge de dades de cada variable o sense cap filtratge en una taula de dades, opció d'exportar i guardar els resultats finals mostrats en la taula en format xls/csv.

3.2.1.1.4. S'exposen les dades de les variables escollides referents als esdeveniments esportius capturats en una gràfica de línies.

Amb les dades filtrades segons els paràmetres previs, i havent escollit les variables de rendiment dels esdeveniments prèviament, es poden representar en una gràfica de línies o de barres per aconseguir un millor anàlisi específic d'un conjunt separat del total de variables de rendiment. D'aquesta manera es podrà identificar els diferents patrons que han seguit els esportistes analitzats i contrastar-los en els de la taula de dades i així, l'analista podrà aconseguir informació que no s'hagi pogut entreveure en la representació de les dades en la taula de dades.

What-Now

El cas d'ús de negoci sense haver implicació de cap tipus de software. L'analista representa les dades obtingudes dels esdeveniments esportius en una gràfica per tenir una visió clara i concisa del rendiment esportiu, quines condicions prèvies hi ha d'haver per la casuística.

BUC: Representar les dades filtrades segons la sèrie temporal prèviament definida en una gràfica lineal.

TRIGGER: Escollir variables de rendiment.

PRECONDICIÓ: Que hi hagin dades, reals, interpolades o en *downsampling*, de variables de rendiment capturades.

IMPLICATS: Analista esportiu.

Acció 1: L'analista estudia les dades de les variables les quals l'hi convenen més segons la seva investigació.

Acció 2: Per a cada variable de rendiment prèviament escollida per l'anàlisi, l'analista traça una gràfica lineal per representar les seves dades. Les gràfiques es van encreuant dins els mateixos eixos.

Acció 3: L'analista contrasta els resultats obtinguts en la gràfica amb els de la taula de dades.

Future-How

El cas d'ús de negoci considerant implicació d'un software operatiu. Quins criteris seguirà l'analista per mostrar les dades de les variables de rendiment des del software hipotètic i implicat en un format de gràfic de dades, quines condicions prèvies hi ha d'haver per la casuística considerada.

PUC: Visualitzar les dades de cada variable de rendiment com a gràfica lineal segons l'estudi que es vol realitzar.

TRIGGER: Seleccionar les variables de rendiment per representar les seves dades en el gràfic.

PRECONDICIÓ: Que hi hagin dades dels esdeveniments esportius pujades i que s'hagin triat les variables de rendiment.

IMPLICATS: Analista esportiu.

Acció 1: Després d'haver enregistrat les variables seleccionades per l'analista, el producte mostra un gràfic de línies, en el qual cada línia representa les dades al llarg de la sèrie temporal, prèviament definida, d'una variable en concret.

OUTCOME: Gràfica lineal de visualització de les dades de les variables de rendiment escollides en la sessió d'anàlisi de rendiment.

3.2.2. User-Stories

Requeriment funcional per a què l'analista seleccioni les variables de rendiment. Funcionalitat important per a l'aplicació i, en conseqüència pel client.

UserStory #: 1.1 **Esdeveniment /PUC:** 1

Descripció: Com analista, vull seleccionar les variables de rendiment.

Raó: Per així, visualitzar i analitzar les dades que necessito durant la sessió d'estudi del rendiment esportiu.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, el 99% de les variables seleccionades s'enregistren de forma correcta, guardant la seva informació corresponent per una posterior representació.

Satisfacció client: 4

Insatisfacció client: 4

Dependències: *Userstory #2.1*

Material de suport: Fitxers csv de dades capturades. (*Data set*)

Requeriment no funcional per la usabilitat, eficiència, eficàcia i rendiment de l'aplicació, partint de la funcionalitat anterior. Agradable pel client. Prescindible per la funcionalitat principal de l'aplicació.

UserStory #: 1.2

Esdeveniment / PUC: 1

Descripció: Com analista, vull seleccionar les variables de rendiment de manera ràpida, eficient i sense molta dificultat.

Raó: Per així, ser efectiu en la selecció de material a analitzar, disminuir el temps en preparar l'estudi o investigació i maximitzar el meu temps de treball.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, 9 de cada 10 professionals selecciona les variables que requereixen per fer l'estudi i ho realitzen en menys de 2 segons.

Satisfacció client: 4

Insatisfacció client: 3

Dependències: *Userstory #1.1*

Material de suport: Fitxers csv de dades capturades. (*Data set*)

Requeriment funcional per a què l'analista introdueixi les dades capturades a l'aplicació recol·lectades en fitxers csv. Funcionalitat important per a l'aplicació i, en conseqüència pel client.

UserStory #: 2.1

Esdeveniment / PUC: 2

Descripció: Com analista, vull introduir les dades de les variables de rendiment capturades.

Raó: Per així, poder filtrar les dades segons un temps i una freqüència de mostreig concrets.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, el 98% de les dades capturades s'enregistren correctament, i el 99% de les filtracions de temps i freqüència es processen sense cap tipus d'error.

Satisfacció client: 5

Insatisfacció client: 5

Dependències: Cap

Material de suport: Fitxers csv de dades capturades. (*Data set*)

Requeriment no funcional per la usabilitat, eficiència, eficàcia i rendiment de l'aplicació, partint de la funcionalitat anterior. Agradable pel client. Prescindible per la funcionalitat principal de l'aplicació.

UserStory #: 2.2

Esdeveniment / PUC: 2

Descripció: Com analista, vull introduir les dades de les variables de rendiment capturades en el menor temps possible i amb la major eficiència.

Raó: Per així, començar el més aviat possible l'anàlisi del rendiment i obtenir resultats el més aviat possible.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, el 98% de les vegades l'analista enregistra els fitxers csv sense cap tipus de problema i rep validació conforme s'han inserit correctament i en un temps màxim de 3 segons.

Satisfacció client: 4

Insatisfacció client: 5

Dependències: *Userstory #2.1*

Material de suport: Fitxers csv de dades capturades. (*Data set*)

Requeriment funcional per a què l'analista estudiï les dades filtrades en un format taula de dades i els resultats se'ls pugui guardar localment. Funcionalitat important per a l'aplicació i, en conseqüència pel client.

UserStory #: 3.1

Esdeveniment / PUC: 3

Descripció: Com analista, vull estudiar, examinar i visualitzar les dades filtrades en una taula de dades i obtenir els resultats finals desitjats.

Raó: Per així, extreure o exportar uns resultats consistents i segons la necessitat de l'anàlisi del rendiment que es realitza.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, el 99% de les dades de cada variable de rendiment es mostren correctament en la taula.

Satisfacció client: 5

Insatisfacció client: 5

Dependències: *Userstory #1.1*

Requeriment no funcional per la usabilitat, eficiència, eficàcia i rendiment de l'aplicació, partint de la funcionalitat anterior. Agradable pel client. Prescindible per la funcionalitat principal de l'aplicació.

UserStory #: 3.2

Esdeveniment / PUC: 3

Descripció: Com analista, vull obtenir les dades en una taula de dades generada en el menor temps possible i que les pugui manipular amb molt poca dificultat.

Raó: Per així, aconseguir els resultats desitjats i tenir una evidència tangible que els verifiqui o per arribar a conclusions de l'anàlisi.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, la taula es genera en menys de 4 segons i el temps d'interacció amb ella per obtenir un cert resultat, un filtre d'una variable per exemple, sigui en menys de 2 segons. Que 8 de cada 10 usuaris obtinguin els resultats que requereixen els seus estudis sense cap complicació i que siguin consistents.

Satisfacció client: 4

Insatisfacció client: 3

Dependències: *Userstories #3.1*

Requeriment funcional per a què l'analista estudiï les dades d'un conjunt de variables de rendiment filtrades i els resultats se'ls pugui guardar localment. Funcionalitat suplementària per a l'aplicació.

User Story #: 4.1

Esdeveniment / PUC: 4

Descripció: Com analista, vull mostrar les dades escollides en una gràfica de línies.

Raó: Per així, descobrir i observar d'una manera més gràfica i visual si hi han hagut variacions o canvis inesperats al llarg de la sèrie temporal en les variables de rendiment en base amb les dades representades en la taula de dades.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, el 97% dels resultats de la taula es visualitzen en la gràfica.

Satisfacció client: 3

Insatisfacció client: 3

Dependències: *User stories #3.1*

Requeriment no funcional per la usabilitat, eficiència, eficàcia i rendiment de l'aplicació, partint de la funcionalitat anterior. Agradable pel client. Prescindible per la funcionalitat principal de l'aplicació.

User Story #: 4.2

Esdeveniment / PUC: 4

Descripció: Com analista, vull que es mostrin les dades escollides prèviament en la gràfica lineal de manera ràpida, eficient i efectiva.

Raó: Per així, contrastar les observacions sobre les dades de la taula i veure el transcurs d'aquestes en tota la sèrie temporal en un temps reduït.

Autor: Analista esportiu

Criteri: El sistema compleix amb el requisit si, en menys de 2 segons, la modificació o filtració que s'ha efectuat en una variable de rendiment dins la taula, es mostri en la gràfica.

Satisfacció client: 3

Insatisfacció client: 2

Dependències: *User stories #4.1*

3.2.3. Prototips

En la **figura 11** es mostra l'apartat de l'aplicació on s'introdueixin els paràmetres de filtratge més adients pel futur anàlisi esportiu, juntament amb la creació de noves sessions d'anàlisi. En aquesta entrada de paràmetres s'enregistraria el temps d'inici en el que es vol començar l'anàlisi de dades, ja sigui el temps en frames o, si es desitja, el temps explícit en hores, minuts, segons i dècimes de segons per filar més prim l'anàlisi. A més a més es seleccionaria també quina és la freqüència en la que es vol treballar durant la sessió d'anàlisi. El botó '+' seria per reiniciar la sessió, és a dir, crear-ne una de nova, i el botó 'next' es confirma l'entrada de paràmetres per l'actual sessió.

The screenshot shows a web application interface with a dark navigation bar at the top containing links: Home, Settings (active), Upload, Data analytics, Chart, Profile, and Log Out. The main content area is white and contains the following elements:

- Initial frame**: A text input field.
- Final frame**: A text input field.
- Initial time**: Four input fields labeled 'h', 'min', 'seg', and 'dec'.
- Final time**: Four input fields labeled 'h', 'min', 'seg', and 'dec'.
- Frequency**: Four buttons labeled '1 f/s' (orange), '5 f/s' (blue), '10 f/s' (green), and '25 f/s' (red).
- Buttons**: A green circular button with a white '+' sign in the top right, and a blue 'Next' button in the bottom right.

Figura 11. Prototip configuració anàlisi.

En la **figura 12** es mostra l'apartat de l'aplicació on es carreguen les dades a analitzar per descobrir el rendiment esportiu de l'atleta o equip esportiu en qüestió. Per a cada fitxer carregat, es donaria un nom que englobaria les variables de rendiment que disposa tal fitxer, per tenir control del que s'està carregant. A més a més, es donaria un temps d'inici i final per filtrar el conjunt de dades dins el fitxer que es vol analitzar, i per finalitzar, la freqüència a la que es vol tenir les dades filtrades per la seva posterior representació.

Home

Statistics

Upload

Data analytics

Chart

Profile

Log Out

Performance variable name

csv File

Initial time

Final time

Frequency

open

Initial time

Final time

Frequency

open

Initial time

Final time

Frequency

open

Initial time

Final time

Frequency

open

Initial time

Final time

Frequency

open

Initial time

Final time

Frequency

open

Next

Figura 12. Prototip carregar fitxers de dades.

En la **figura 13** es mostra l'apartat de l'aplicació on es visualitzin les dades manipulades i s'apliqui un nou filtratge en elles per a obtenir un anàlisi exhaustiu. Amb opció d'aconseguir capturar l'anàlisi realitzat i emmagatzemar-lo localment. La primera columna de la taula de dades es mostraria el temps per a cada valor de cada variable de rendiment representada, i les columnes adjacents es visualitzarien els valors de cadascuna relacionats amb el seu temps de captura. Cada columna tindria un filtratge segons una condició especificada relacionada amb la columna de dades en qüestió, per afinar encara més l'anàlisi.

Home

Settings

Upload

Data analytics

Chart

Profile

Log Out

Temporal set	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11
Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
13.57.55	Value 1	Value 1	Value 1	Value 1	Value 1	Value 1	Value 1	Value 1	Value 1	Value 1	Value 1
13.57.56	Value 2	Value 2	Value 2	Value 2	Value 2	Value 2	Value 2	Value 2	Value 2	Value 2	Value 2
13.57.57	Value 3	Value 3	Value 3	Value 3	Value 3	Value 3	Value 3	Value 3	Value 3	Value 3	Value 3
13.57.58	Value 4	Value 4	Value 4	Value 4	Value 4	Value 4	Value 4	Value 4	Value 4	Value 4	Value 4
13.57.59	Value 5	Value 5	Value 5	Value 5	Value 5	Value 5	Value 5	Value 5	Value 5	Value 5	Value 5
13.58.00	Value 6	Value 6	Value 6	Value 6	Value 6	Value 6	Value 6	Value 6	Value 6	Value 6	Value 6

Export xls / csv

Figura 13. Prototip taula anàlisi de dades.

En la **figura 14** es mostra l'apartat de l'aplicació per seleccionar les variables de rendiment trobades segons els fitxers de dades carregats en l'aplicació. Es selecciona les variables de rendiment de tots els fitxers carregats disponibles per a la seva posterior visualització en la taula de dades.

Display performance data

- ☐ Variable 1
- ☒ Variable 2
- ☒ Variable 3
- ☐ Variable 4
- ☒ Variable 5
- ☐ Variable 6
- ☐ Variable 7
- ☐ Variable 8
- ☒ Variable 9
- ☒ Variable 10
- ☐ Variable 11
- ☒ Variable 12
- ☐ Variable 13
- ☐ Variable 14
- ☒ Variable 15

Save

Variables Selector

Figura 14. Prototip selector de variables de rendiment.

3.2.4. Refinament dels requeriments

Durant les fases de disseny i implementació s'han modificat els requisits que el client prèviament demanava. Bàsicament s'han anat ajustant a mesura que es desenvolupava l'aplicació.

Aleshores els canvis substantius respecte els requeriments inicials són:

1. No es necessita definir un rang de temps de filtratge (sèrie temporal) per a cada fitxer de dades de rendiment ni una freqüència.
2. S'ha suprimit el filtratge de dades per a cada variable de rendiment (filtre en cada columna de la taula de dades, tampoc en la columna de temps)
3. S'introduiran paraules clau segons cada csv enregistrat per tal de què es processin les dades correctament.
4. Les dades de les variables de rendiment dels esdeveniments esportius que es vulguin representar en el gràfic de dades no és necessari que s'hagin transformat mitjançant processos d'interpolació o downsampling.

Les mesures per complir aquest refinament s'explica més endavant en les consideracions de la implementació.

Llavors es conserven els requeriments tals que:

1. Un selector per seleccionar les variables de rendiment de les quals es vol analitzar les seves dades.
2. Una configuració de paràmetres de filtratge per la visualització de la taula de dades, tals com una freqüència de mostreig per les dades i un rang de temps per filtrar la quantitat de dades a representar en la taula
3. Una taula de dades per representar les dades dels fitxers amb les seves variables de rendiment seleccionades prèviament amb l'opció de guardar els resultats localment.
4. Un gràfic de dades per representar les dades de les variables de rendiment escollides prèviament amb l'opció de guardar els resultats localment.

CAPÍTOL IV

Fase de desenvolupament

4. Presa de decisions per al desenvolupament del software

4.1. *Framework* i llenguatges

Django

El *framework* Web de Python d'alt nivell *Django* [8] motiva a construir aplicacions web d'una manera pragmàtica, neta i ràpida. S'ocupa de la dificultat del desenvolupament Web, és per això que no hi ha necessitat de reinventar res. És lliure i de codi obert. Dissenyat per crear des de zero aplicacions el més ràpid i eficient possible. S'encarrega de la seguretat i ajuda als desenvolupadors a evitar problemes de seguretat. També es caracteritza per la seva gran escala de flexibilitat.

Empreses, organitzacions i governs han fet ús de *Django* per construir tota mena d'aplicacions, des de sistemes de gestió a xarxes socials i a plataformes científiques. Amb aquest *framework* es pot desenvolupar el *back-end* (capa del servidor), en Python, i el front-end (capa de presentació), en CSS i HTML.

4.2. Bases de dades

La base de dades que suporta per defecte *Django* és *SQLite 3* [32], la qual serà tractada durant tot el desenvolupament del Projecte. És una llibreria del llenguatge C que implementa un mort de bases de dades totalment equipat, alta fiabilitat, i de processament ràpid. El seu format de fitxers és estable, multi-plataforma i retro-compatible. És lliure i el seu codi font és públic. També es

podria fer ús de *PostgreSQL* donada l'especificació d'aquesta en el *DockerFile* del Projecte.

Django gestiona els models de l'aplicació junt amb la base de dades per a què sigui consistent, és a dir, que satisfaci les restriccions d'integritat (unicitat de la clau, integritat referencial, domini, clau no nul·la).

4.3. Control de versions

L'eina de control de versions que s'ha utilitzat durant el desenvolupament de l'aplicació web és GitHub. També s'ha vinculat el control de versions de desplegues de *Heroku* amb el repositori de GitHub emprat durant la implementació.

4.4. Esquema de desplegament

Ús de *Docker* [1] com a contenidor software de virtualització on pugui córrer l'aplicació software sense tractar amb la màquina local. Aquesta eina funciona com una màquina virtual sense poc pes en memòria. La seva funcionalitat principal és automatitzar el desplegament d'aplicacions, proporcionant una nova capa d'abstracció en diferents sistemes operatius.

Remarcar que en aquest projecte no s'ha pogut executar l'entorn de *Docker*, ja que en el sistema operatiu Microsoft Windows es necessita la versió de Windows 10 Pro o Windows 10 Enterprise. Tot i així es facilita els arxius necessaris per executar una contenidor virtual *Docker*.

Ús de *Heroku* per desplegar la plataforma web en un dels seus repositoris remots com a servidor al núvol.

4.5. Patrons de disseny utilitzats

El patró de disseny *Model-View-Template* (MVT) és el que respecta *Django*, per tant, és l'utilitzat en el Projecte del Treball. Aquest patró deriva del patró de disseny *Model-View-Controller* (MVC) [16].

L'objectiu principal és separar el model del domini de l'aplicació (l'estructura central), la manera com es presenta el model a l'usuari (gràficament, visualment) i la manera en què l'usuari interactua amb aquest (la interfície entre el model, la vista i els inputs de dispositius externs) [19].

- El model s'encarrega de la base de dades. És una capa d'accés de dades, conté els camps i comportaments de les dades que es vol emmagatzemar. Un model és una classe en Python i no coneix res sobre les altres capes de *Django*.
- La vista executa la lògica de negoci i interactua amb el model per portar les dades a la representació via plantilla. Captura les dades d'un model. A més a més dona a la plantilla accés a dades per visualitzar-les o simplement les processa.
- La plantilla és la capa de presentació que s'encarrega de la interfície de l'usuari. Utilitzant codi HTML es representen les dades.

El patró Singleton s'utilitza amb l'objectiu de restringir la instanciació del model de configuració per a la taula de dades de l'aplicació. També es considerat un anti-patró, ja que varies vegades s'utilitza com a eufemisme de variable global.

4.6. Diagrames de disseny

4.6.1. Diagrama de classes

En el diagrama que es mostra en la **Figura 15**, els models de dades de la base de dades de l'aplicació web. Remarcar que un usuari podrà carregar a la base de dades més d'una mostra de rendiment, és a dir, font de dades de rendiment i podrà tenir una sola configuració per l'anàlisi de la taula de dades (aquí és on actuarà el patró de disseny *Singleton*). *Django* et dóna per defecte el model *Usuari* amb els camps, atributs i mètodes ja implementats.

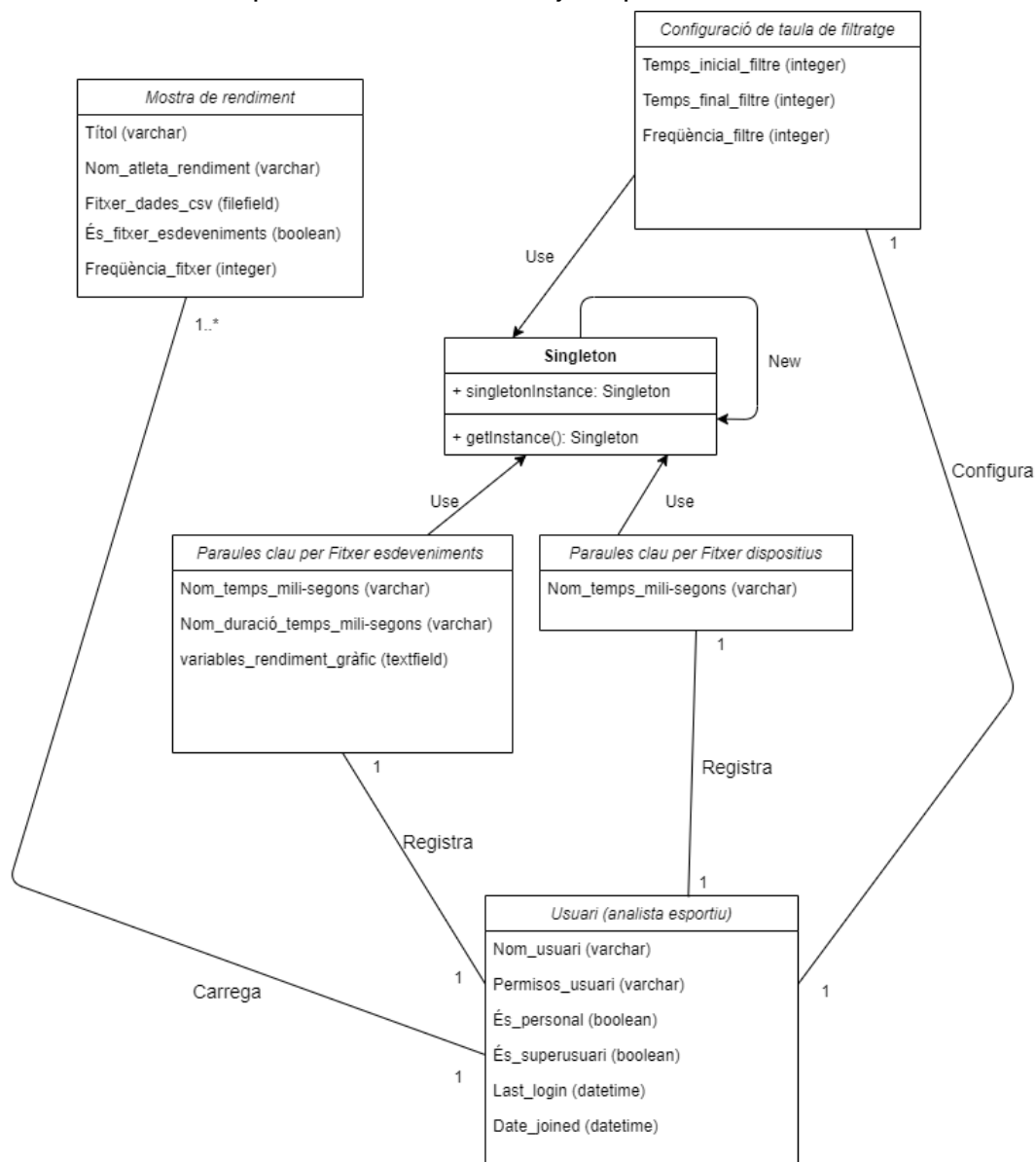


Figura 15. Diagrama de classes UML

4.6.2.Diagrama de flux de treball

La funcionalitat principal de l'aplicació web es reflecteix en la **Figura 16**. L'usuari si no està registrat, s'haurà de crear un nou compte i es guardarà en la base de dades de l'aplicació.

Primerament, s'ha de comprovar si no hi ha cap fitxer guardat en la base de dades d'alguna sessió anterior o si no s'ha carregat cap mai. Si no hi ha fitxers carregats, l'aplicació demana a l'usuari que carregui un o més fitxers amb el format de guardat corresponent, en aquest cas, format csv.

Un cop carregat/s, es necessita introduir les paraules clau útils pel processament intern del programa. L'aplicació mostra una llista amb totes les paraules necessàries per a què es puguin guardar les paraules clau necessàries segons els fitxers que s'han carregat.

Seguidament es comprova si hi ha una configuració establerta d'altres sessions o si s'ha de crear una de nova. Si es així, s'introdueixen els paràmetres per configurar la sessió d'anàlisi corresponent.

A continuació, l'usuari té tot el que cal per a què pugui començar el seu anàlisi esportiu. Quan acaba el seu estudi, pot o bé exportar les dades que ha analitzat o si li convé carregar més fitxers de dades per aprofundir el seu anàlisi.

-

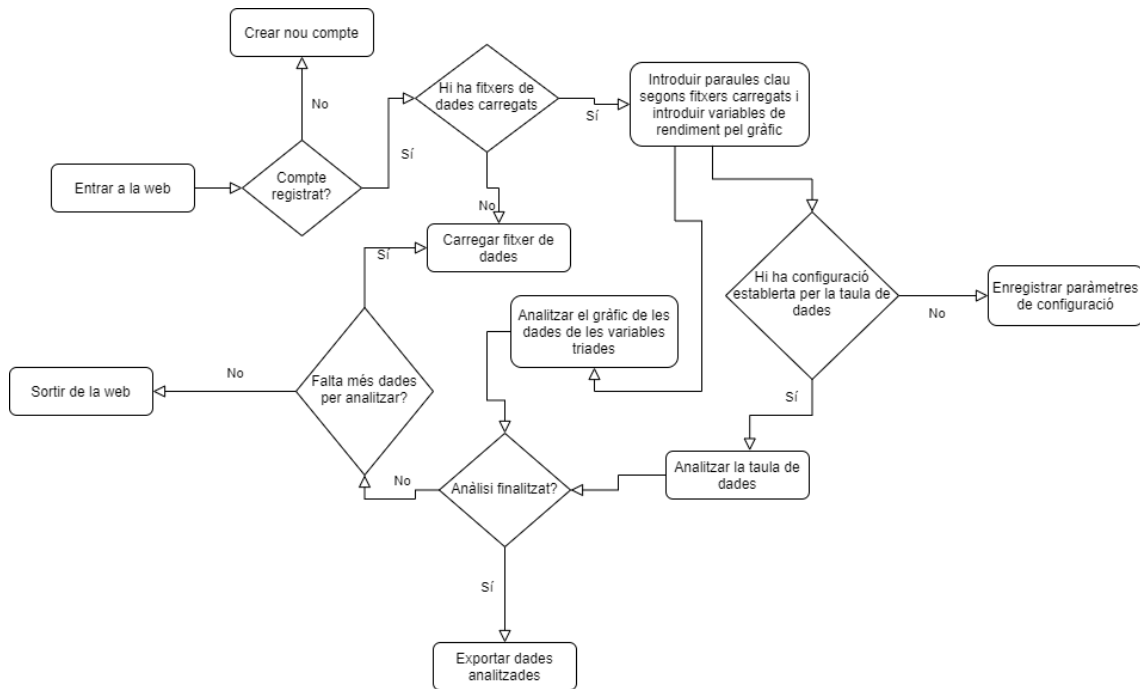


Figura 17. Diagrama de flux de treball

4.7. Implementació

4.7.1. Organització del Projecte software

Un directori general, on es guarden els sub-directoris i fitxers de configuració de desplegament i de gestió de l'aplicació, entre altres, anomenat 'app'. Dins de 'app' un directori per les migracions de la base de dades, les modificacions que s'han fet sobre aquesta; un directori per fitxers estàtics: el fitxer d'estil de la capa de presentació de la web, imatges per l'estètica de la web; un directori per les plantilles HTML, els fitxers Html per cadascuna de les vistes de negoci; un directori pels fitxers del projecte explicats a continuació.

Aquesta organització de directoris ve donada gràcies a les bones pràctiques del *framework Django*.

4.7.1.1. Fitxers del Projecte (directori TFG)

Els fitxers per la configuració general del projecte en *Django*, per la gestió de la publicació de la web al servidor remot, per la gestió de les migracions i altres menesters de la base de dades, etc.

- **manage.py**: Permet interactuar amb el projecte de *Django* de moltes maneres i, entre altres coses, ajuda a l'administració del lloc web, inicia un servidor web en la màquina local i fixa les configuracions del fitxer `settings.py`
- **settings.py**: Conté totes les configuracions de la web i per la instal·lació del *Django*.
- **urls.py**: Conté les rutes que estan disponibles en el projecte. *Django* permet dissenyar el *routing* de la web com es vulgui, sense limitacions en el *framework*. Així, es pot construir un esquema de urls elegant i polit.
- **wsgi.py**: Configuració que s'utilitza per connectar-se al sistema en producció. Suporta la Web Server Gateway Interface (convenció de crides als servidors web per enviar peticions a les aplicacions web o *frameworks*)
- **asgi.py**: per desplegar en servidors web i aplicacions asíncrones.

4.7.1.2. Fitxers de l'aplicació (directori app)

Fitxers on es pot desenvolupar tota la lògica necessària per aconseguir crear la web. Amb ells s'implementen els models de dades de la base de dades, els formularis per capturar les dades introduïdes per l'usuari, i tota la lògica de negoci de les vistes de la web, entre altres.

- **models.py**: Les representacions de les dades de l'aplicació. Cada classe de cada model conté els camps bàsics i el comportament de les dades que seran emmagatzemades.

- **admin.py:** Per a registrar els models creats en l'aplicació i per a què la base de dades els tingui en compte a l'hora de migrar els camps de dades.
- **apps.py:** Ajuda a incloure qualsevol configuració de l'aplicació. S'emmagatzema metadades de l'aplicació
- **forms.py:** Els formularis per capturar les dades que insereix l'usuari i que es guarden en els models respectius, és a dir, es guarden en la base de dades.
- **urls.py:** Routing de l'aplicació web.
- **views.py:** Tota la lògica de negoci per donar resposta a una sol·licitud. Funcions que reben una petició i responen amb un contingut.
- **templates (directori):** Reben informació de la vista per mostrar el resultat d'una manera organitzada (fitxers HTML).

4.7.1.3. Fitxers de desplegament

Fitxers necessaris pel desplegament de la web, tan en el servidor de la web com en el contenidor virtual.

- **Dockerfile:** Document de text que conté totes les ordres que un usuari podria cridar a la línia de comandes per muntar una imatge docker. Amb aquest arxiu, Docker crea un nou contenidor, com si s'hagués executat una comanda de crear un contenidor docker manualment. Docker assigna un sistema de fitxers de lectura en escriptura al contenidor, com a capa final. Això permet que un contenidor en execució creï o modifiqui fitxers i directoris al seu sistema de fitxers local.
- **docker-compose.yml:** eina per definir i executar aplicacions Docker per a diversos contenidors. S'utilitza aquest arxiu per configurar els serveis de l'aplicació. Llavors, amb una única comanda, es crea i s'inicia tots els serveis des de la configuració establerta.

- **Procfile:** per especificar a **Heroku** el servidor web de l'aplicació. Simplement es declara el tipus de procés, en aquest cas el servidor web que es vol utilitzar pel desplegament al núvol de l'aplicació software.

El fitxer *requeriments.txt* conté tots els paquets, és a dir, totes les dependències del Projecte pel seu bon funcionament i execució.

4.7.2. API's externes utilitzades en el Projecte software.

4.7.2.1. Python Data AnalysisLibrary – *pandas*

La API *pandas* [20] és una llibreria de codi obert, llicenciada per la BSD, que aporta alt rendiment i facilitat d'ús en les estructures de dades i les eines d'anàlisi de dades pel llenguatge de programació Python. Aquesta llibreria té com a dependència la API *numpy* (llibreria per computació científica en Python, referent a MATLAB).

Amb *pandas* [34] es pot realitzar, entre altres:

- Fusionar i unir conjunts de dades.
- Alinear dades i tractar amb les dades que falten.
- Manipulació de dades mitjançant indexació integrada d'objectes DataFrame.
- Generació d'interval de dades, conversió de freqüència, canvi de dates, retard i altres funcionalitats de sèries horàries. Anàlisi de sèries horàries.
- Lectura de fitxers amb CSV, XLSX, TXT, entre altres formats.

En la implementació de la web, s'utilitza aquesta API per a llegir els fitxers **csv** de fila en fila, i per cada fila es va tractant cada variable de rendiment i la seva dada corresponent a la fila. Els fitxers és emmagatzemats en la base de dades durant la sessió de l'usuari. En finalitzar tot el recorregut per les files del fitxer, ja es pot començar amb la interpolació de les dades guardades.

4.7.2.2. Python *math* Idem

Aquest mòdul de Python proveeix de funcions matemàtiques definides per l'estàndard de C. Puntualitzar que aquestes funcions no suporten nombres complexos [30].

Math pot donar, entre altres, funcions tals com:

- Representació teòrica de nombres (ceil, factorial, isnan etc.)
- Potències i logaritmes
- Trigonometria
- Conversió angular

En la implementació de l'aplicació web, s'utilitza aquesta web per a convertir els valors de les columnes llegides dels **csv** que són NaN (Not a Number) en valor 'null'.

4.7.2.3. Python *json* Idem

El mòdul de Python *json* [29] serveix principalment per codificar i/o decodificar objectes en Python a objectes que poden ser tractats en *JavaScript* i al inrevés. Més informació sobre el que ofereix aquest mòdul en la documentació de Python. Tanmateix, en el desenvolupament de l'aplicació web, s'ha fet ús del mètode *dumps* per codificar objectes en Python a *JavaScript*.

4.7.3. Consideracions durant la implementació

Seguidament es mostraran les diferents consideracions més importants que s'han tingut en compte en la fase d'implementació de l'aplicació web.

Primer de tot l'apartat dels paràmetres de configuració per a la sessió de l'usuari es recopilen amb el formulari que es mostra en la **Figura 17**: temps inicial i final en mili-segons per l'anàlisi de les dades final; camp de la freqüència de la taula

de dades en què es representaran les dades dels fitxers amb la seva freqüència equivalent a la de la taula.

Init time ms*

Time in milliseconds.

Fin time ms*

Time in milliseconds.

Frequency*

5 Hz

Data table frequency.

Next

Figura 17. Formulari paràmetres de configuració

Per entrar els noms clau que s'utilitzaran dins el *backend* per gestionar les dades, gràcies als dos formularis per entrar noms clau, un referent a fitxers d'esdeveniments, com es mostra en la [Figura 18](#), en aquest també hi ha la opció d'enregistrar les variables de rendiment, les dades de les quals es vulguin representar en el gràfic de dades, i un altre per les claus dels fitxers de dispositius, com es mostra en la [Figura 19](#).

Time ms name*

Column name of Time in milli-seconds values.

Duration time ms name*

Column name of Time Duration in milli-seconds values.

Chart perf vars

Performance variables separated by comma for showing their data to the chart data. Use white spaces if you like.

Next

Figura 18. Formulari, per fitxers esdeveniments, de noms clau i noms de les variables de rendiment per representar en el gràfic de dades.

Time name*

Column name of Time of DEVICES files.

Next

Figura 19. Formulari noms clau per fitxers de dispositius

S'ha optat per llistar els fitxers de dades que s'aniran carregant al llarg de la sessió, com es mostra en la **Figura 20**, per així, tenir un control de quines dades es van a analitzar. Cadascun d'ells tindrà una opció d'eliminar-se de la base de dades, en cas que s'hagi carregat per equivocació o, simplement, que es vulgui tractar un altre fitxer diferent.

Upload csv file

Set events file key words

Set devices file/s key words

Title	Athlete	CSV File	Delete
		csv_files/1_AC_S2.csv	Delete
		csv_files/1_AC_S2_acelero.csv	Delete
gps	jugador 1	csv_files/1_AC_S2_GPS.csv	Delete

Figura 20. Llista de fitxers carregats durant la sessió

Per a carregar un fitxer, s'ha especificat el formulari que es mostra en la **Figura 21**. Remarcar que els camps de 'Title' i 'Athlete' són opcionals, són per donar un nom de referència a quines dades s'han carregat.

Title

Athlete

Csv*

Seleccionar archivo

Ningún archivo seleccionado

Event file*

Yes

Is it events file what you are uploading?

Frequency*

Ignore the field when uploading EVENTS file.

Upload

Figura 21. Formulari per carregar fitxer de dades

Primerament, es volia separar el selector de variables de rendiment i la taula de dades, però, per motius de pura lògica de la codificació i per comoditat del desenvolupador, aquestes dues funcionalitats s'han unit en una mateixa pàgina web, tal i com es mostra en la **figura 22**.

☒ Tmilisegundos
 ☐ TFrames
 ☐ DuracionFr
 ☐ TSegundos
 ☐ DuracionSeg
 ☐ DuracionMiliseg
 ☐ Enfrontament
 ☐ Situacio
 ☐ Ronda
 ☐ Jugadors_Equip_A_Groc
 ☐ Jugadors_Equip_C_Blau
 ☐ Accio
 ☐ Zona
 ☐ TIME
 ☐ WIMU_1__ACEL_X(G)_
 ☐ WIMU_1__ACEL_Y(G)_
 ☐ WIMU_1__ACEL_Z(G)_
 ☐ WIMU_2__ACEL_X(G)_
 ☒ WIMU_2__ACEL_Y(G)_
 ☐ WIMU_2__ACEL_Z(G)_
 ☐ WIMU_3__ACEL_X(G)_
 ☐ WIMU_3__ACEL_Y(G)_
 ☐ WIMU_3__ACEL_Z(G)_
 ☐ WIMU_4__ACEL_X(G)_
 ☐ WIMU_4__ACEL_Y(G)_
 ☐ WIMU_4__ACEL_Z(G)_
 ☐ WIMU_5__ACEL_X(G)_
 ☐ WIMU_5__ACEL_Y(G)_
 ☒ WIMU_5__ACEL_Z(G)_
 ☐ WIMU_1__GPS_LAT(degrees)_
 ☐ WIMU_1__GPS_LON(degree)_
 ☐ WIMU_2__GPS_LAT(degrees)_
 ☐ WIMU_2__GPS_LON(degree)_

Tmilisegundos	WIMU_2__ACEL_Y(G)_	WIMU_4__ACEL_X(G)_	WIMU_5__ACEL_Z(G)_
79040	-0,763506532	0,320620686	-0,355758101
79240	0,419065297	-0,045139771	-1,082725286
79440	-1,076313853	1,001297951	-0,063905336
79640	0,360631883	-0,157466382	-0,306365907
79840	-0,157779112	1,250801802	-1,005951643
80040	-0,360786736	-0,080876969	0,242788598
80240	-0,398382306	1,478537202	-0,163185716
80440	0,168996617	-0,179249838	-0,574925303
80640	-0,068900064	0,959370613	-1,215950847
80840	0,055633634	1,036598325	-0,129977494
81040	0,110179462	-0,217485562	-0,025115712

Figura 22. Taula de dades + selector de variables

Per concordar les sèries temporals de ambdós tipus de fitxers, s'ha tingut en compte que les dades del fitxer d'esdeveniments es tractin d'acord amb el camp dels mili-segons. A més a més, que en els fitxers de dades capturades per dispositius externs, se li assignin el valor en mili-segons del primer temps de captura de dades del fitxer d'esdeveniments a la primera captura del valor de temps menor entre aquests fitxers.

També, el valor de les files de la columna de temps per als fitxers dels dispositius, s'incrementarà en 1 fins arribar a la longitud marcada pel fitxer. Llavors, per a cada dada de cada variable de rendiment en cada mili-segon del fitxer d'esdeveniments es relacionarà amb la dada amb el mateix mili-segon de cada fitxer de dispositiu.

Quan ja no es puguin relacionar els mili-segons, és a dir, que en els fitxers de dades dels dispositius no hi hagi més valors capturats, l'últim valor relacionat serà per igual a la resta de captures que queden per relacionar en el fitxer.

S'ha considerat que el programa elimini els accents dels fitxers de dades, i en el cas dels fitxers d'esdeveniments, a l'hora de filtrar les dades amb els temps inicial i final de l'apartat de configuració de la taula de dades, es canviï de posició la columna que conté els mili-segons i posar-la en la primera columna, per a què l'algoritme de filtratge de dades faci efecte, mostrat en la **figura 23**.

```
def filter_time_files(dict_down_sampled_files, init_filter_time, fin_filter_time, events_time_name,
                     events_duration_time_name, devices_time_name):
    files_to_render = []
    for file in dict_down_sampled_files:
        df = pd.DataFrame.from_dict(file, orient="columns")
        df.to_csv("filtered_time_files.csv")
        csv = pd.read_csv("filtered_time_files.csv", header=0, index_col=[0])
        os.remove("filtered_time_files.csv")
        performance_variables = csv.columns.values.tolist()
        data = {}
        for var in performance_variables:
            data[var] = []
        for row in csv.values.tolist():
            filter_time = False
            for (element_row, element_perf_var) in zip(row, performance_variables):
                if element_perf_var == events_time_name or element_perf_var == devices_time_name:
                    if fin_filter_time >= element_row >= init_filter_time:
                        filter_time = True
                        data[element_perf_var].append(element_row)
                else:
                    if filter_time:
                        data[element_perf_var].append(element_row)
            float_data_to_int_data(data, events_duration_time_name)
            files_to_render.append(data)
    return files_to_render
```

Figura 23. Algoritme de filtratge de dades segons configuració de usuari

S'ha considerat que alguns fitxers poden tenir valors en coma flotant i que sorgeixin conflictes a l'hora de mostrar-los en la taula. Per tant, es mostra en la **Figura 24** l'algoritme per tractar els coma flotant i convertir-los a enters.

```

def float_data_to_int_data(csv_dict, events_duration_time_name):
    keys_floats = {}
    """Handle float numbers except nan"""
    for key in csv_dict.keys():
        for value in csv_dict[key]:
            if isinstance(value, float):
                if key not in keys_floats.keys() and not math.isnan(value):
                    keys_floats[key] = []
                if key in keys_floats.keys():
                    if math.isnan(value):
                        if key == events_duration_time_name:
                            keys_floats[key].append(0)
                        else:
                            keys_floats[key].append(value)
                    else:
                        keys_floats[key].append(int(math.floor(value)))
            else:
                keys_floats[key].append(int(math.floor(value)))

    for key in keys_floats:
        csv_dict[key] = keys_floats[key]
    """Handle float numbers except nan"""

```

Figura 24. Algorisme conversió de float a enter

Per facilitar el procés de l'aplicació, s'ha considerat transformar la freqüència dels fitxers de dades a la més alta que hi pot haver, és a dir, 1000 Hz, amb el mètode interpolació, el qual es mostra en l'algorisme de la [Figura 25](#), per als fitxers d'esdeveniments, i en la [figura 26](#) per als fitxers de dispositius.

```

def interpol_events(data, csv, perf_vars, time_lasting, events_time_name, events_duration_time_name):
    for row, time in zip(csv.values.tolist(), time_lasting):
        for (element_row, element_perf_var) in zip(row, perf_vars):
            if isinstance(element_row, int) and element_perf_var == events_time_name:
                extreme_time_value_to_interpolate = element_row
                # Interpolate time
                for new_time in range(extreme_time_value_to_interpolate, time + extreme_time_value_to_interpolate):
                    data[element_perf_var].append(new_time)
            else:
                # Interpolate str value
                for str_value in range(time):
                    data[element_perf_var].append(element_row)
    float_data_to_int_data(data, events_duration_time_name)

```

Figura 25. Interpolació de fitxers d'esdeveniments


```
def interpol_devices(data, csv, perf_vars, interpol_value, limit_length):
    extreme_time_value_to_interpolate = 0
    for row in csv.values.tolist():
        for (element_row, element_perf_var) in zip(row, perf_vars):
            if isinstance(element_row, int):
                extreme_time_value_to_interpolate = element_row
            if not (extreme_time_value_to_interpolate + interpol_value > limit_length):
                if not isinstance(element_row, int):
                    for str_value in range(interpol_value):
                        # Interpolate value str
                        data[element_perf_var].append(element_row)
                if isinstance(element_row, int):
                    # Interpolate time
                    for time in range(extreme_time_value_to_interpolate,
                                      extreme_time_value_to_interpolate + interpol_value):
                        data[element_perf_var].append(time)
```

Figura 26. Interpolació de fitxers de dispositius

Per a la representació de les dades només ha calgut utilitzar el mètode **downsampling** mostrat en l'algoritme de la **figura 27** i filtrar les dades segons els paràmetres de configuració de l'usuari amb el mètode especificat en la **figura 23**.

```
def down_sample(dict_csv, table_frequency, events_time_name, devices_time_name):
    # Frequency of dict_csv data is 1000 Hz
    if table_frequency != 1000:
        average = int(round(1000/table_frequency))
        time = []
        for key in dict_csv.keys():
            downsampled = dict_csv.get(key)[0::average]
            if key == events_time_name or key == devices_time_name:
                for element in downsampled:
                    time.append(round(element/10)*10)
                dict_csv[key] = time
            else:
                dict_csv[key] = downsampled
    return dict_csv
```

Figura 27. Downsampling

A l'hora de representar les dades, s'ha utilitzat l'algorisme en JavaScript que es mostra en la **figura 28**. En aquest, s'han recuperat de la capa del backend mitjançant funcions *json*, i s'han tractat les dades de manera que es poguessin representar en una taula HTML.

```

function generateDataTable(columnNames, columnValues) {
    let body = document.body, tbl = document.createElement('table');
    let subListLength = 0;

    let tr = tbl.insertRow();
    for (let i = 0; i < columnNames.length; i++) {
        let th = document.createElement("TH");
        th.appendChild(document.createTextNode(columnNames[i].toString()));
        tr.appendChild(th);
    }

    for (let value in columnValues){
        if (columnValues.hasOwnProperty(value)) {
            subListLength = columnValues[value].length;
            break
        }
    }

    for (let i = 0; i < subListLength; i++){
        let tr = tbl.insertRow();
        for (let j = 0; j < columnValues.length; j++){
            let td = tr.insertCell();
            td.appendChild(document.createTextNode(columnValues[j][i].toString()));
            columnValues[j].shift();
        }
    }
    body.appendChild(tbl);
}

```

Figura 28. Algorisme representació dades en la taula html

A l'hora de representar les dades de les variables de rendiment escollides, s'ha utilitzat l'algorisme en JavaScript que es mostra en la [figura29](#). En aquest, s'han recuperat de la capa del backend mitjançant funcions *json*, i s'han tractat les dades de manera que es poguessin representar en un o més tipus de gràfics gràcies al mètode per crear un gràfic amb els seus respectius paràmetres segons la [figura30](#) del projecte Chart.js.

```

let lists_labels = [];
let lists_data = [];
let chart_vars = [];
let contexts = [];

chart_vars = {{ chart_vars|safe }};
for (let i = 0; i < chart_vars.length; i++)
    contexts.push(document.getElementById(chart_vars[i]).getContext('2d'));
lists_data = {{ lists_data|safe }};
lists_labels = {{ lists_labels|safe }};
// contexts.length = lists_labels.length = lists_data.length = chart_vars.length

let category = getCheck();
for (let i = 0; i < contexts.length; i++){
    setChart(contexts[i], category, lists_labels[i], lists_data[i], chart_vars[i], getColors(lists_labels[i]));
    document.querySelector("." + chart_vars[i]).addEventListener("click", function () {
        exportDataToCSV(chart_vars[i]+".csv", lists_data[i], lists_labels[i]);
    });
}

```

Figura 29. Algorisme per crear el gràfic de dades

```

function setChart(context, category, labels, data, label, backColor) {
    new Chart(context, {
        // The type of chart we want to create
        type: category,

        // The data for our dataset
        data: {
            labels: labels,
            datasets: [{
                label: label,
                backgroundColor: backColor,
                borderColor: backColor,
                data: data
            }]
        },

        // Configuration options go here
        options: {
            scales: {
                yAxes: [{
                    ticks: {
                        beginAtZero: true
                    }
                }]
            },
            legend: {
                display: false
            }
        }
    });
}

```

Figura 30. Creació d'un gràfic gràcies a Chart.js

CAPÍTOL V

Fase de validació

Els objectius d'aquesta fase són demostrar qualitat o comportament apropiat del software, detectar i solucionar problemes. La idea del *testing* suposa la execució del software i la observació del seu comportament o el seu resultat de sortida. Si es troba una fallada, s'ha de localitzar i arreglar el desperfecte que ha causat la fallada. El propòsit del *software testing* és assegurar que els sistemes software funcionin com s'espera quan siguin utilitzats pels clients i usuaris a qui van dirigits.

La finalitat d'aquesta fase és gestionar una bona millora i qualitat del software per a què així, la següent fase de manteniment sigui el més lleugera possible, i a més a més, s'incrementi la capacitat del producte per satisfer les necessitats implícites (els requeriments) quan es faci un ús en unes condicions específiques.

5.1. Tests del Projecte

S'han testat les funcionalitats principals que ha de tenir l'aplicació implementades en les vistes i la capa de presentació representada en els fitxers HTML, a base del mètode Ad-hoc que s'explica a continuació. Les vistes es troben el fitxer *views.py*

5.1.1. Ad-hoc Testing

La manera més simple de validar les funcionalitats de la plataforma ha estat executar el software i observar, per comprovar el comportament esperat del sistema. Aquest mecanisme simple de test s'anomena *ad-hoc* [13]. És un tipus de *testing* informal amb l'objectiu de trencar el funcionament del sistema. No segueix cap estructura de *testing* i es realitza aleatòriament en qualsevol part de

l'aplicació. La finalitat principal d'aquest mètode és trobar errors, utilitzant la tècnica "Error endevinat". A l'hora de validar el software, s'ha tingut bon coneixement del producte de negoci i per tant, un coneixement profund dels requisits de l'aplicació

No obstant, si es valida el software repetidament utilitzant aquest mètode, es necessita seguir el que s'ha realitzat en les validacions anteriors, i així evitar malgastar el temps repetint els mateixos tests. Per fer aquest seguiment només ha calgut, per exemple, una *checklist* per anar comprovant quines funcionalitats s'anaven validant i revisant el bon funcionament de la web per obtenir, així, un software de qualitat.

5.1.2. **Checklists per la validació i qualitat del Software**

Checklist de revisió del disseny de la Base de Dades

Per tal de garantir qualitat en la base de dades de l'aplicació, es revisarà l'arquitectura de la base de dades gràcies a la *checklist* de la **Taula 2**.

Pregunta	Sí	No	N/A	Nota
La base de dades és adequadament normalitzada?	X			
Estan tots els atributs requerits en el disseny?	X			
La clau primària és correcta en cada entitat?	X			
Les entitats estan normalitzades en funció de la seva clau principal?	X			
Les relacions entre les entitats són correctes?	X			
Comprovació de relacions circulars	X			
El disseny físic correspon amb el lògic?	X			
La mida i el tipus dels camps estan correctament especificats?	X			
Estan els índexs necessaris creats en les taules?			X	

S'ha comprovat l'efecte del SGBD sobre el disseny?	X			
El SGBD pot encarregar-se del volum de dades requerit?	X			
S'ha comprovat l'impacte de rendiment degut a índexs alternatius?		X		
El disseny de la base de dades és prou flexible per permetre afegir nous camps en el futur?	X			
Comprovació d'estàndards de codificar la base de dades			X	

Taula 2. Checklist validació qualitat de la base de dades

No es té coneixement de codificació de la base de dades per tant no s'ha realitzat cap tipus de correcció per aquesta detecció. Django, per defecte no proveeix de índexs necessaris per les taules ni índexs alternatius, per tant, no s'ha contemplat aquesta deficiència en el desenvolupament. Tot i així són mancances que afecten a la bona qualitat del software.

Checklist de funcionament intern correcte de la web

Per tal d'anar comprovant el bon funcionament de l'aplicació web es farà ús d'una *checklist* tal com figura en la **Taula 3**.

Ordre etapes	Descripció etapa	Resultat esperat	Èxit o fallada
1	Registrar nou compte + contrasenya.	Nou compte + contrasenya xifrada en la base de dades.	Èxit
2	Log in amb nou compte i nova contrasenya.	S'entra al contingut de la web.	Èxit
3	Anar a "Settings".	Es mostra el formulari de configuració.	Èxit

4	Omplir el formulari per la configuració de la sessió d'anàlisi (temps inici, final i freqüència camps obligatoris).	En acabar es re-direcciona a la pàgina d'inici.	Èxit
5	Comprovar que s'ha guardat la configuració exacta en la base de dades.	S'emmagatzema correctament.	Èxit
6	Anar a "Files".	Es mostra l'historial de fitxers (buit) i botó de pujar fitxer	Èxit
7	Es veu l'historial de fitxers pujats buit.	No hi ha fitxers pujats.	Èxit
8	Click al botó "Uploadcsvfile"	Es mostra el formulari de pujar fitxer.	Èxit
9	Omplir formulari per pujar un fitxer csv amb algunes dades generals per la seva descripció (tots els camps obligatoris).	En acabar es re-direcciona a la pàgina de l'historial de fitxers pujats.	Èxit
10	Comprovar que s'han guardat les dades del fitxer i aquest mateix en la base de dades.	S'emmagatzema correctament.	Èxit
11	En l'historial de fitxers pujats comprovar que hi ha el fitxer pujat anteriorment.	Es guarda en l'historial correctament.	Èxit
12	Si és el cas, click a "Delete" per esborrar-lo de l'historial i de la base de dades.	S'esborra de l'historial.	Èxit
13	Comprovar que el fitxer s'ha esborrat de la base de dades.	S'esborra també de la base de dades.	Èxit
14	Les dades dels fitxers es guarden segons: nom de la columna (variable de rendiment, temps inclòs) com a clau i els valors dins la columna com a valor del diccionari.	Un diccionari per a cada fitxer csv carregat amb claus iguals als noms de les columnes i amb valors iguals a llistes amb les	Èxit

		dades de cada columna.	
15.1	Si hi ha fitxer d'esdeveniments carregat, comprovar que, en la columna de temps en el fitxer de dispositiu, es comenci amb el primer temps de la columna dels mili-segons del fitxer d'esdeveniments i anant incrementant el valor fins a arribar a una freqüència de mostreig igual a 1000 Hz, és a dir, es va incrementant el valor a 1.	La columna de temps en els fitxers de dispositius començant amb el primer valor de la columna de mili-segons dels esdeveniments i incrementant-se fins a arribar a la mida de la freqüència	Èxit
15.2	Si no hi ha fitxer d'esdeveniments carregat, comprovar que, en la columna de temps en el fitxer de dispositiu, es comenci amb 0 mili-segons i es vagi incrementant el valor igual que en 15.1	La columna de temps en els fitxers de dispositius començant amb el primer valor igual a 0 i incrementant-se fins a arribar a la mida de la freqüència	Èxit
16	Comprovar que es van interpolant els valors de les variables de rendiment del/s fitxers de dispositius segons convingui el tipus de dades.	Per a cada valor de temps hi ha un valor de variable de rendiment.	Èxit
17	Comprovar que la columna de temps del fitxer d'esdeveniments es vagi omplint de valors per arribar a la freqüència de 1000 Hz	La mida de la llista de valors de temps és igual a la dels diccionaris dels fitxers de dispositius.	Èxit
18	Comprovar que es van interpolant els valors de les variables de rendiment del/s fitxers d'esdeveniments segons convingui el tipus de dades.	La mida de les llistes del diccionari arriba a la estimada, igual a la de les llistes dels	Èxit

		diccionaris dels fitxers de dispositius.	
19	Una vegada interpolats tots els fitxers, comprovar que el <i>downsample</i> redueix la freqüència, és a dir, es redueix la mida de cada llista de dades per a cada variable de rendiment	S'arriba a una reducció equivalent en totes les llistes de valors de cada diccionari.	Èxit
20	Comprovar que els noms de les variables de rendiment, les dades de les quals es volen analitzar, es mostren per ser seleccionades en la pàgina.	Es mostren les variables per a ser seleccionades.	Èxit
21	Comprovar si en seleccionar variables i guardar la selecció es mostra la taula amb les columnes amb el nom de les variables seleccionades.	Es mostra la taula amb les columnes amb el nom de les variables seleccionades.	Èxit
22	Comprovar que realment funciona el patró Singleton a l'hora de crear una única configuració per la taula.	Només es crea una configuració per la taula.	Èxit
23	Comprovar que, amb la configuració de la taula, es mostren les dades filtrades de les variables seleccionades.	Es mostren les dades filtrades de les variables seleccionades, segons la configuració establerta.	Èxit
24	Comprovar que s'exporta en un fitxer csv les dades que es mostren en la taula de dades.	S'exporta en un fitxer csv les dades que es representen en la taula.	Èxit

25	Comprovar que es guarden en la base de dades els noms de les variables de rendiment, les dades de les quals es volen representar en el gràfic de dades	Es guarden tots els noms en un string en la base de dades.	Èxit
26	Comprovar que s'obtenen les dades de les variables de rendiment que es volen mostrar en el gràfic de dades	Es guarden correctament en un diccionari de manera que el nom de la variable fa de clau i les seves dades de valor.	Èxit
27	Comprovar que es mostra les opcions per triar de quin tipus de gràfic es vol utilitzar en la sessió, i segons la opció triada, es mostra un tipus de gràfic o un altre.	Es mostren totes les opcions dels diferents tipus i segons cadascuna es mostra un gràfic diferent.	Èxit
28	Comprovar que es mostren les dades, de les variables de rendiment escollides prèviament, en el gràfic.	Es mostren les dades per cada variable escollida en l'ordre en el que estan en el fitxer.	Èxit
29	Comprovar que s'exporta en un fitxer csv les dades representades en el gràfic.	S'exporten les dades del gràfic en qüestió en un fitxer csv.	Èxit

Taula 3. Checklist validació funcionalitats de la web

CAPÍTOL VI

Conclusions

6.1. Conclusions finals

D'ençà que es va començar el Treball ha evolucionat adaptant-se a problemes que sorgien i a necessitats noves fins a la seva finalització. Com a resultat, s'ha aconseguit una aplicació software útil per realitzar un anàlisi complet de les dades de rendiment de l'esport, facilitant així la cerca d'una substancial millora en l'esport. Aquesta eina facilita una visualització de les dades clara i concisa, fent possible l'obtenció d'un coneixement precís del rendiment dels atletes, a fi de, a partir d'anàlisis exhaustius sobre aquestes dades trobar els anomenats patrons de moviment i de comportament i, juntament amb aquests, perfeccionar la pràctica esportiva.

Tot i que només s'ha extret dades d'un sol tipus de pràctica esportiva, amb el mateix equipament específic utilitzat en aquesta extracció, s'hagués pogut efectuar l'extracció de dades en qualsevol altre esport. Tant és així, que l'aplicació està dissenyada per suportar qualsevol conjunt de dades sempre amb el mateix o similar format en el que s'estructuren els fitxers d'emmagatzemament de dades que s'han utilitzat durant el desenvolupament i validació de l'aplicació.

Donat que la idea de crear aquest software va ser originada pel Dr. Àngel Ric de l'INEFC de Lleida, actualment s'espera integrar els resultats derivats d'aquest projecte amb el *Lince*: **multiplatform sport analysis software**, de la Universitat Politècnica de Catalunya. Aquesta integració seria a mans dels experts en esports col·lectius de l'INEFC.

Seguidament, les conclusions que puc extreure havent acabat aquest Treball Final de Grau, respecte els objectius que vaig proposar són:

1. Des del meu punt de vista, desenvolupar **un Backend satisfactori** i òptim, no és una tasca senzilla. S'ha de tenir molta cura en el treball amb la base de dades local i, especialment, amb la remota, ja que, en aquest aspecte és on més m'han aparegut complicacions de tota mena. En canvi, la lògica de negoci, si no es podia fer d'una forma, es podia d'una altra.
2. En relació al *Frontend* de l'aplicació, en la meva opinió, el llenguatge JavaScript ha jugat un paper clau en la funcionalitat principal de la capa de presentació i en la meva àrea de coneixement degut a l'aprenentatge de la estructura, semàntica i significat.
3. En conseqüència de l'estudi realitzat, referent a l'avantguarda de l'esport, considero de vital importància el gran efecte que tenen les tecnologies de l'esport en la millora del rendiment i la pràctica esportiva. Des del meu punt de vista, considero que s'hauria d'emfatitzar més el suport d'aquestes tecnologies, no només en l'esport d'elit, sinó també en l'esport de la ciutadania, per tal de millorar la condició física de la societat.
4. L'anàlisi de rendiment o qualsevol anàlisi s'ha de realitzar sobre una representació de les dades a analitzar nítida, intel·ligible, precisa i rigorosa, doncs aquest anàlisi porta a la fita que es vol arribar en l'estudi o recerca principal, per tal de trobar els patrons de comportament, o, simplement descobrir l'objecte de l'anàlisi, dels subjectes a investigar.
5. Per a gestionar qualsevol tipus de dades es necessita tenir un bon pla d'acció, és a dir, executar un exhaustiu estudi de les principals fonts d'origen d'aquestes dades i esbrinar detalladament quina és la seva organització interna, a fi d'aconseguir els millors resultats possibles enfront l'objectiu de la gestió.
6. Un cop acabada la fase de requeriments del procés de desenvolupament del software, me n'adono que es produeix una gran diferència entre el que el client d'una aplicació vol primerament per implementar i el que realment

s'acaba desenvolupant, sempre amb la conformitat que dona el mateix client, és clar, però, a més a més i a grans trets, aquesta realitat es dona en l'àmbit professional de les solucions informàtiques d'avui en dia.

6.2. Resultats

El resultat d'aquest projecte, des del començament, ha sigut analitzar i estudiar el comportament col·lectiu i patrons de comportament entre els diferents integrants d'un equip esportista col·lectiu o un atleta.

6.3. Treball futur

Partint com a base la funcionalitat principal de la web desenvolupada, existeixen diversos punts del projecte que es podrien millorar, com ara:

- Enregistrament de paraules clau per al tractament de fitxers d'esdeveniments: Actualment es tracta amb un sol fitxer d'esdeveniments per sessió, d'acord amb la necessitat del projecte, per això s'entra un sol conjunt de paraules clau per un sol fitxer d'aquest tipus. Cabria la possibilitat de què, en un moment donat, es volgués tractar dos fitxers d'esdeveniments a la vegada, en una mateixa sessió, per tant, una solució òptima seria, a l'hora de carregar el fitxer d'esdeveniments, enregistrar les seves paraules clau pel seu posterior tractament en la lògica de negoci de l'aplicació.
- Nou filtratge en cada variable de rendiment de la taula de dades: Per aprofundir més l'anàlisi de les dades en la funcionalitat de la taula de l'aplicació, escauria la possibilitat de crear un filtre per a cada columna (és a dir per a cada variable), i així filtrar segons el contingut de qualsevol variable i, havent realitzat el filtratge, representar les dades de les altres variables que el seu contingut concordés (estigués en la mateixa fila) amb

el/s valor/s filtrat/s; de la mateixa manera que es pot contemplar en un dels prototips inicials construïts per al projecte actual.

Per altra banda convindria esmentar algunes noves funcions per enriquir més el contingut de la web les quals s'haurien de construir des de zero:

- Virtual Player [3]: un gràfic en 2D que, amb les dades recollides dels fitxers, monitoritzi la pràctica esportiva gràcies a les dades les quals s'han extret d'aquesta (exemple: les jugades que s'han desenvolupat de l'esport col·lectiu respectiu). Juntament amb aquest gràfic, es mostraria un mapa de calor amb les zones afectades del camp d'extracció de les dades. Aquesta eina estaria més focalitzada en l'anàlisi en grup d'un esport col·lectiu.
- Àrea d'anàlisi de posició i velocitat [3]: un gràfic de *Kagi* [14] per representar la velocitat de l'atleta o qualsevol altra mesura física segons les dades reunides. Aquesta funció faria èmfasi en l'anàlisi individual esportiu. Addicionalment, un gràfic de línies per a mostrar variables i/o mètriques pròpies de l'esport col·lectiu.

Bibliografía

- [1] Anderson, C. (2015). Docker [Software engineering]. IEEE Software, 102 - c3.
- [2] Becker, A. (s.f.). *Kalman Filter*. Obtenido de <https://www.kalmanfilter.net/default.aspx>
- [3] Benito Santos, A., Theron, R., Losada, A., Sampaio, J., & Lago Peñas, C. (2018). Data-Driven Visual Performance Analysis in Soccer: An Exploratory Prototype. *Frontiers in Psychology*.
- [4] Cano, J. (2020). Angular: Mucho más que un framework. SG.
- [5] Cengage. (13 de Marzo de 2020). *Sport Performance*. Obtenido de <https://www.encyclopedia.com/sports/sports-fitness-recreation-and-leisure-magazines/sport-performance>
- [6] D, L., D, K., & M, L. (2018). Validation of electronic performance and tracking systems EPTS under field conditions. *PLOS ONE*.
- [7] Direct, S. (s.f.). *Movement Pattern*. Obtenido de <https://www.sciencedirect.com/topics/medicine-and-dentistry/movement-pattern>
- [8] Django. (2020). *Django Project*. Obtenido de <https://www.djangoproject.com/>
- [9] Duarte, R., Araujo, D., Folgado, H., Tiago Esteves, P., Marques, P., & Davids, K. (2013). Capturing complex, non-linear team behaviours during competitive football performance. *ResearchGate*.
- [10] Evenson, K. R., & Terry, J. W. (2012). Assessment of Differing Definitions of Accelerometer Nonwear Time. *PubMed Central*.
- [11] Frencken, W., Lemmink, K. A., Delleman, N., & Visscher, C. (2011). Oscillations of centroid position and surface area of soccer teams in small-sided games. *ResearchGate*.
- [12] Gil Bellosta, C. (2018). *R para profesionales de los datos: una introducción*.
- [13] guru99. (2020). *What is Adhoc Testing? Types with Example*. Obtenido de <https://www.guru99.com/adhoc-testing.html>
- [14] IngenioVirtual. (2020). *Tipos de gráficos y diagramas para la visualización de datos*. Obtenido de <https://www.ingeniovirtual.com/tipos-de-graficos-y-diagramas-para-la-visualizacion-de-datos/>
- [15] Iñigo, J. A. (10 de Marzo de 2017). ¿Qué es la Programación Reactiva? Una introducción. *¿Qué es la Programación Reactiva? Una introducción*.

- [16] Joy, A. (2020). *Difference Between MVC and MVT Patterns*. Obtenido de <https://pythonistaplanet.com/difference-between-mvc-and-mvt/>
- [17] Kaiser, T., Zheng, F., & Dimitrov, E. (2009). An Overview of Ultra-Wide-Band Systems With MIMO. *Proceedings of the IEEE*, 285-312.
- [18] Kirmizibayrak, C., Honorio, J., Jiang, X., Mark, R. and Hahn, J.K., 2011. Digital analysis and visualization of swimming motion. *International Journal of Virtual Reality*, 10(3), pp.9-16.
- [19] Krasner, G., & Pope, S. (1988). A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System. *JOOP - Journal of Object-Oriented Programming*.
- [20] McKinney, W. (12 de Enero de 2020). *pandas*. Obtenido de <https://pandas.pydata.org/>
- [21] Mecafenix, F. (2017). Giroscopio sensor de velocidad angular. *Ingeniería Mecafenix*.
- [22] Media, O. (2020). *Mastering Shiny*.
- [23] Moya Ramón, M., Reina Vaillo, R., Gutiérrez Aguilar, O., Vera García, F., López Elvira, J., Aracil Marco, A., & Paredes Ortiz, J. (2007). Nuevas tecnologías aplicadas al desarrollo y control del entrenamiento y la competición en el deporte. *Revista de la Facultad de Ciencias Sociales y Jurídicas de Elche*, 156-183.
- [24] Oliden, P. E. (2009). ¿Existe vida más allá del SPSS? Descubre R. *Psicothema*, 652-655.
- [25] Omoregie, P. (2016). The impact of technology on sport performance. *Research Gate*.
- [26] Pincus, S. (1995). Approximate entropy (ApEn) as a complexity measure. *ResearchGate*.
- [27] Platzi. (2019). *Frontend con Angular*. Obtenido de <https://platzi.com/desarrollo-angular/>
- [28] Project, N. (2020). *Node.js*. Obtenido de <https://nodejs.org/en/>
- [29] Python. (2020). *json - JSON encoder and decoder*. Obtenido de <https://docs.python.org/3/library/json.html>
- [30] Python. (2020). *math - Mathematical functions*. Obtenido de <https://docs.python.org/3/library/math.html>
- [31] Robertson, S., & Robertson, J. (2012). *Mastering the requirements process*.
- [32] SQLite. (2020). *What is SQLite?* Obtenido de <https://www.sqlite.org/index.html>

- [33] Sumpter, D. (2017). *Using heat maps to assess attacking play*. Obtenido de <https://medium.com/@Soccermatics/using-heat-maps-to-assess-attacking-play-56348f52dee5>
- [34] Team, D. (2018). *Python Pandas Tutorial – Learn Pandas in Python (Advance)*. Obtenido de <https://data-flair.training/blogs/pandas-tutorial/>
- [35] Wattles, M. (10 de Marzo de 2019). *Sports Performance Metrics: Why the Numbers Matter!* Obtenido de <https://edgemeridian.com/sports-performance-metrics/>
- [36] White, S., Walley, K., Johnston, J., Henderson, P., Hale, K., Andrews, W. J., & Siann, J. (2003). *USA Patente nº 6531982B1*.